

# **AUTOMATIC GENERATION OF TRADING RULES ADJUSTED TO THE MARKET STATE**

**Carlos Martín Fernández**

En cumplimiento parcial de los requisitos para el grado de Doctor en

**CIENCIA Y TECNOLOGÍA INFORMÁTICA**

Universidad Carlos III de Madrid

Directores:

Dr. Pedro Isasi Viñuela  
Dr. David Quintana Montero

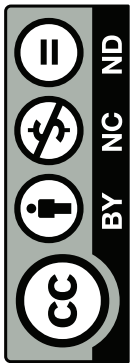
Tutor:

Dr. David Quintana Montero

Leganés. junio, 2019

ISBN: 978-84-09-14557-7

*Esta página se ha dejado en blanco intencionadamente.*



© Carlos Martín Fernández. Some rights reserved.

This work is distributed under Creative Commons 4.0 license. You are free to copy, distribute and transmit the work under the following conditions:

- (i) You must give appropriate credit and provide a link to the license in any reasonable manner, but not in any way that suggests the licensor endorses you or your use;
- (ii) You may not use the material for commercial purposes;
- (iii) If you remix, transform, or build upon the material, you may not distribute the modified material. You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits.

See <http://creativecommons.org/licenses/by-nc-nd/4.0/> for further details.

*Esta página se ha dejado en blanco intencionadamente.*

# **AUTOMATIC GENERATION OF TRADING RULES ADJUSTED TO THE MARKET STATE**

**Carlos Martín Fernández**

En cumplimiento parcial de los requisitos para el grado de Doctor en

**CIENCIA Y TECNOLOGÍA INFORMÁTICA**

Universidad Carlos III de Madrid

Directores:

Dr. Pedro Isasi Viñuela  
Dr. David Quintana Montero

Tutor:

Dr. David Quintana Montero

Leganés. junio, 2019

ISBN: 978-84-09-14557-7

*Esta página se ha dejado en blanco intencionadamente.*

# Agradecimientos

Debo agradecer de manera especial y sincera al profesor David Quintana Montero su apoyo y confianza en mi trabajo. Su capacidad para guiar mis ideas ha sido un aporte inestimable, facilitándome siempre la información que necesitaba y leyendo incansablemente todos los correos electrónicos y borradores de esta trabajo. Al profesor Pedro Isasi Viñuela por ser el artífice, a través de sus publicaciones, de que haya cursado este doctorado.

El agradecimiento más profundo y sentido para mi familia. Sin su apoyo, colaboración e inspiración habría sido imposible. A mi mujer Alicia, a mis hijos Andrés, Jara, Manuel y Samara en los que pienso todo el tiempo. Pero sobre todo a mi madre a quien no puedo olvidar.

*Esta página se ha dejado en blanco intencionadamente.*



# Publicaciones relacionadas

C. Martín, D. Quintana y P. Isasi.

**Autor.**

**Title:** Evolution of trading strategies with flexible structures: a configuration comparison [135].

**Date:** Nov 2018.

**Journal:** *Neurocomputing*.

**Volume:** 331.

**Pages:** 242-262.

**I.F. 2017:** 3.241 | **Q1** Neurocomputing.

**doi:** 10.1016/J.NEUCOM.2018.11.062 <https://doi.org/10.1016/j.neucom.2018.11.062>

**url:** <https://www.sciencedirect.com/science/article/pii/S0925231218314061?via%3Dihub>

**ISSN:** 0925-2312.

**Publisher:** Elsevier.

**Incluida parcialmente en la tesis.**

**Capítulo** Selección de la técnica apropiada de computación evolutiva.

*Esta página se ha dejado en blanco intencionadamente.*

# Abstract

The search for profitable trading strategies has been driving research efforts for decades. Among the many different approaches that can be used to design trading rules (such as Particle Swarm Optimization, Genetic Algorithms, Artificial Neural Networks and Fuzzy Methods), there are some based on evolutionary computation that are especially interesting due to three key features: the process of rule generation is automatic, the resulting rules are interpretable, and their structure is flexible.

Since Allen and Karjalainen published their seminal piece on evolution of trading rules using Genetic Programming (GP), many authors have made related contributions either based on the same technique, or Grammatical Evolution (GE).

Most of these contributions generate investment rules based on a combination of raw market data and technical indicators and, unlike related approaches that use genetic algorithms or evolution strategies to optimize predefined rules, these have the advantage of creating flexible structures automatically.

A common limitation is that it is often the case that the approaches are static and do not take into account the structural changes of the state of the market. Given that this phenomenon is very prevalent in financial time series, the decision rules are commonly derived from market environments that do not hold in test periods.

The problem of adjusting to structural changes is that we must choose between two opposite extremes: keeping the same model over time, or updating it constantly. Even though the second might seem, at least in principle, more appropriate, there is a possibility that the constant change in the model will have undesirable consequences due to transaction costs. The evolutionary process of GP/GE considers commissions throughout the period as part of the fitness function, and that makes it select rules that generate a limited number of signals. However, it

is possible that a constant model update interferes with that endogenous control mechanism of the number of purchase and sale orders.

This Tesis tackles with dynamic trading system solutions based on the use of ensembles and GE. The approach combines the possibility of changing the model as a reaction to changes in the price generation mechanism, with an inertia component that mitigates the consequences of overtrading. We also work with a different approach that is not based on ensembles but on a system that takes advantage of an internal hysteresis mechanisms that is part of the own models.

# Índice general

<b>Índice general</b>	<b>IX</b>
<b>Índice de figuras</b>	<b>XV</b>
<b>Índice de Tablas</b>	<b>XIX</b>
<b>1 Introducción</b>	<b>1</b>
1.1 Contexto . . . . .	1
1.2 Definición del problema . . . . .	3
1.3 Hipótesis de trabajo . . . . .	11
1.4 Objetivos . . . . .	12
1.5 Metodología . . . . .	12
1.6 Estructura del documento . . . . .	14
<b>2 Estado del arte</b>	<b>19</b>
2.1 Contribuciones que tratan la generación automática de reglas de inversión . . . . .	19
2.1.1 Contribuciones basadas en programación genética . . . . .	19
2.1.2 Contribuciones basadas en variantes de la programación genética . . . . .	29
2.1.3 Contribuciones basadas en gramáticas evolutivas . . . . .	31
2.2 Contribuciones basadas en sistemas dinámicos . . . . .	35

<b>3</b>	<b>Computación evolutiva</b>	<b>41</b>
3.1	Programación genética . . . . .	41
3.1.1	Representación . . . . .	43
3.1.2	Inicialización . . . . .	46
3.1.3	Operadores . . . . .	50
3.1.4	Conjuntos de elementos terminales y no terminales . . . . .	63
3.1.5	Función de aptitud . . . . .	64
3.1.6	Elitismo . . . . .	66
3.1.7	Control de tipos . . . . .	66
3.1.8	Funciones definidas automáticamente . . . . .	70
3.1.9	Condiciones de parada . . . . .	71
3.1.10	Parámetros de ejecución de un algoritmo de PG . . . . .	73
3.1.11	Parsimonia y crecimiento del código . . . . .	77
3.2	Gramáticas evolutivas . . . . .	79
3.2.1	Introducción . . . . .	79
3.2.2	Representación . . . . .	80
3.2.3	Inicialización . . . . .	81
3.2.4	Operadores . . . . .	86
3.2.5	Control de tipos . . . . .	90
3.3	Extensiones comunes a la PG y las GE . . . . .	91
3.3.1	La búsqueda de la solución en PG y GE . . . . .	91
<b>4</b>	<b>Aspectos generales de la solución</b>	<b>99</b>
4.1	Entorno de trabajo . . . . .	100
4.2	Representación de las reglas de inversión . . . . .	101

---

4.2.1	Conjunto de funciones no terminales y elementos terminales	102
4.2.2	Gramática . . . . .	104
4.3	Función de aptitud . . . . .	105
4.4	Parametrización . . . . .	110
4.4.1	Parametrización en la PG . . . . .	112
4.4.2	Parametrización de las GE . . . . .	113
4.4.3	Generación de aleatoriedad . . . . .	115
<b>5</b>	<b>Metodología de evaluación</b>	<b>117</b>
5.1	Elección del tamaño de la muestra . . . . .	117
5.2	Protocolo de evaluación de significación estadística . . . . .	119
5.2.1	Intervalo de confianza . . . . .	119
5.2.2	Contraste de hipótesis . . . . .	121
<b>6</b>	<b>Selección de la técnica apropiada de computación evolutiva</b>	<b>133</b>
6.1	Introducción . . . . .	133
6.2	Conjuntos de datos y diseño experimental . . . . .	134
6.3	Resultados experimentales . . . . .	136
6.3.1	Experimento base . . . . .	136
6.3.2	Análisis de sensibilidad paramétrica . . . . .	143
6.3.2.1	Conjunto de indicadores técnicos extendidos . . . . .	144
6.3.2.2	Población extendida . . . . .	151
6.3.2.3	Conjunto de indicadores técnicos y población extendidos . . . . .	153
6.4	Resumen y conclusiones . . . . .	159

<b>7</b>	<b>Enfoque de conjunto de modelos en inversión algorítmica</b>	<b>163</b>
7.1	Introducción . . . . .	163
7.2	Descripción del sistema . . . . .	165
7.3	Análisis experimental . . . . .	170
7.3.1	Conjunto de datos y diseño experimental . . . . .	170
7.3.2	Parametrización . . . . .	172
7.3.3	Resultados experimentales . . . . .	174
7.4	Resumen y conclusiones . . . . .	180
<b>8</b>	<b>Selección dinámica de modelos</b>	<b>183</b>
8.1	Introducción . . . . .	183
8.2	Análisis experimental . . . . .	188
8.2.1	Conjuntos de datos y diseño experimental . . . . .	188
8.2.2	Parametrización . . . . .	189
8.2.3	Resultados experimentales . . . . .	190
8.3	Resumen y conclusiones . . . . .	196
<b>9</b>	<b>Solución de selección dinámica vs. conjuntos de modelos</b>	<b>199</b>
9.1	Introducción . . . . .	199
9.2	Análisis . . . . .	200
<b>10</b>	<b>Conclusiones y trabajos futuros</b>	<b>205</b>
<b>11</b>	<b>Contribución de esta tesis</b>	<b>213</b>
<b>Anexo A</b>	<b>Selección de técnicas apropiadas de CE</b>	<b>215</b>
A.1	Mejores estrategias de inversión obtenidas en la configuración base .	215
A.2	Gráficos de las mejores estrategias de inversión obtenidas en la configuración base . . . . .	219



---

<b>Anexo B</b>	<b>Protocolo de Pruebas</b>	<b>223</b>
<b>Anexo C</b>	<b>Selección técnica apropiada de CE</b>	<b>227</b>
C.1	Configuración base 500 individuos . . . . .	227
C.1.1	Resultados de las simulaciones . . . . .	227
C.1.2	Estadísticos . . . . .	230
C.1.3	Contrastes estadísticos . . . . .	231
C.2	Configuración población ampliada 3000 individuos . . . . .	232
C.2.1	Resultados de las simulaciones . . . . .	232
C.2.2	Estadísticos . . . . .	235
C.2.3	Contrastes estadísticos . . . . .	236
C.3	Configuración conjunto de funciones ampliado . . . . .	237
C.3.1	Resultados experimento . . . . .	237
C.3.2	Estadísticos . . . . .	240
C.3.3	Contrastes estadísticos . . . . .	241
C.4	Configuración conjunto de funciones y población ampliada 3.000 individuos . . . . .	242
C.4.1	Resultados de las simulaciones . . . . .	242
C.4.2	Estadísticos . . . . .	245
C.4.3	Contrastes estadísticos . . . . .	246
C.5	Coste computacional . . . . .	247
<b>Glosario</b>		<b>249</b>
<b>Acrónimos</b>		<b>261</b>
<b>Bibliografía</b>		<b>269</b>

*Esta página se ha dejado en blanco intencionadamente.*

# Índice de figuras

3.1	Regla de inversión de ejemplo, con su representación en árbol correspondiente. . . . .	45
3.2	Comportamiento de la regla (Or ( $>M \times 2$ M2) (Or ( $>$ Mínimo UR) ( $<$ M10 LR))) en 2007. . . . .	45
3.3	Tabla de selección de individuos por el sistema de ruleta. . . . .	55
3.4	Modelo de probabilidad de selección de individuos por el sistema de ruleta. . . . .	56
3.5	Modelo de probabilidad de selección de individuos por el sistema de ranking. . . . .	57
3.6	Modelo de probabilidad de selección de individuos por el sistema de torneo determinista de dos individuos. . . . .	59
3.7	Modelo de probabilidad de selección de individuos por el sistema de torneo probabilístico de dos individuos. . . . .	61
3.8	Árbol de tipos en programación genética. . . . .	69
3.9	Ejemplo de función de PG definida automáticamente. . . . .	70
3.10	Ejemplo de regla de inversión con ADF. . . . .	71
3.11	Construcción de un individuo en GE. . . . .	84
3.12	Cruce de cromosomas en las GE por un punto. . . . .	86
3.13	Cruce de cromosomas en las GE por dos puntos. . . . .	87
3.14	Cruce uniforme de cromosomas en las GE. . . . .	88

4.1	Función de aptitud (Fitness). . . . .	109
6.1	Comportamiento de las mejores estrategias configuración línea base. Entrenamiento: 2004-2006, prueba: 2007. Ge y GePA. . . . .	138
6.2	Distribución de resultados del experimento base por configuración. . . . .	139
6.3	Coste computacional de los experimentos por configuración. . . . .	143
6.4	Comportamiento de la regla “gePa” (And (Mayor Media2 Media1) (Or (Menor LR S1) (Menor Media6 Cierre))) en 2011. . . . .	158
6.5	Comportamiento de la regla “ge” (And (Menor Min1 Media12) (And (Mayor Media2 Media1) (Or (Menor LR Minimo) (Not (Or (And (Me- nor Min1 Maximo) (Mayor Media3 Minimo)) (And (Mayor Cierre S1) (Mayor Min1 LR)))))) en 2011. . . . .	159
7.1	Flujo de trabajo del sistema de conjuntos. . . . .	166
7.2	Sistema de ventanas en la generación de reglas. . . . .	167
7.3	Ilustración del comportamiento del conjunto de modelos. . . . .	169
8.1	Sistema de selección dinámica de modelos de inversión. . . . .	186
8.2	Ilustración del comportamiento del sistema dinámico de modelos. . . . .	187
A.1	Comportamiento de las mejores estrategias configuración línea base. Entrenamiento: 2004-2006, prueba: 2007. . . . .	220
A.2	Comportamiento de las mejores estrategias configuración línea base. Entrenamiento: 2008-2010, prueba: 2011. . . . .	221
A.3	Comportamiento de las mejores estrategias configuración línea base. Entrenamiento: 2012-2014, prueba: 2015. . . . .	222
C.1	Resultados 2007. Configuración base (Entrenamiento 2004-06), 500 individuos. . . . .	227
C.2	Resultados 2011. Configuración base (Entrenamiento 2008-10), 500 individuos. . . . .	228

---

C.3	Resultados 2015. Configuración base (Entrenamiento 2012-14), 500 individuos. . . . .	229
C.4	Estadísticas configuración base 500 individuos. . . . .	230
C.5	Contrastes estadísticos configuración base 500 individuos. . . . .	231
C.6	Resultados 2007. Población ampliada 2007 (Entrenamiento 2004-06), 3.000 individuos. . . . .	232
C.7	Resultados 2011. Población ampliada (Entrenamiento 2008-10), 3.000 individuos. . . . .	233
C.8	Resultados 2015. Población ampliada (Entrenamiento 2012-14), 3.000 individuos. . . . .	234
C.9	Estadísticas configuración extendida 3.000 individuos. . . . .	235
C.10	Contrastes estadísticos configuración extendida 3.000 individuos. . .	236
C.11	Resultados 2007. Funciones ampliadas (Entrenamiento 2004-06), 500 individuos. . . . .	237
C.12	Resultados 2011. Funciones ampliadas (Entrenamiento 2008-10), 500 individuos. . . . .	238
C.13	Resultados 2015. Funciones ampliadas (Entrenamiento 2012-14), 500 individuos. . . . .	239
C.14	Estadísticas conjunto de funciones ampliado 500 individuos . . . . .	240
C.15	Contrastes conjunto de funciones ampliado 500 individuos. . . . .	241
C.16	Resultados 2007. Funciones ampliadas (Entrenamiento 2004-06), 3.000 individuos. . . . .	242
C.17	Resultados 2011. Funciones ampliadas (Entrenamiento 2008-10), 3.000 individuos. . . . .	243
C.18	Resultados 2015. Funciones ampliadas (Entrenamiento 2012-14), 3.000 individuos. . . . .	244
C.19	Estadísticas configuración extendida 3.000 individuos. . . . .	245
C.20	Contrastes conjunto de funciones ampliado 3.000 individuos. . . . .	246

C.21 Coste computacional en función del número de simulaciones. . . . .	247
C.22 Coste computacional en función del número de individuos. . . . .	247
C.23 Coste computacional en función del número de generaciones. . . . .	248

# Índice de Tablas

2.1	Trabajos relacionados. Estrategias y técnicas de CE utilizadas. . . . .	34
3.1	Gramática usada en la regla de inversión de la figura. 3.1. . . . .	83
4.1	Gramática de una regla de inversión. . . . .	104
4.2	Ampliación de una gramática con la definición de una ADF. . . . .	105
4.3	Definición gramatical de una ADF. . . . .	105
4.4	Parametrización de la PG usada por Koza. . . . .	110
6.1	Resumen de resultados del experimento base. . . . .	137
6.2	Significación estadística del experimento base, de las diferencias en las muestras de prueba. . . . .	137
6.3	Resultados del experimento base vs. Buy-and-Hold. . . . .	140
6.4	Análisis de sensibilidad al aumento del tamaño de la población y del conjunto de indicadores técnicos. . . . .	144
6.5	Análisis de sensibilidad al aumento de indicadores técnicos vs. experimento base. . . . .	149
6.6	Resultados del aumento de indicadores técnicos vs. Buy-and-Hold. . . . .	150
6.7	Análisis de sensibilidad al aumento de la población vs. experimento base. . . . .	151
6.8	Resultados del aumento de la población vs. Buy-and-Hold. . . . .	152

6.9	Análisis de sensibilidad al aumento de la población e indicadores técnicos vs. experimento base. . . . .	153
6.10	Resultados del aumento de población e indicadores técnicos vs. Buy-and-Hold. . . . .	154
6.11	Resumen del análisis de sensibilidad. . . . .	155
6.12	Promedios de rentabilidad vs. B&H de todas las configuraciones. . .	156
7.1	Parametrización de los tests de conjuntos. . . . .	173
7.2	Retorno neto de las estrategias de conjuntos de modelos. . . . .	175
7.3	Número de transacciones en estrategias de conjunto. . . . .	177
7.4	Retorno bruto de las estrategias de conjuntos. . . . .	179
8.1	Retorno neto en la comparativa de la estrategia dinámica. . . . .	191
8.2	Número de transacciones estrategia dinámica. . . . .	193
8.3	Retorno bruto en la comparativa de la estrategia dinámica. . . . .	195
9.1	Retorno neto en la comparativa entre modelo basado en Inercia y el Dinámico. . . . .	200
9.2	Número de transacciones comparativa entre modelo basado en Inercia y el Dinámico. . . . .	201
9.3	Retorno bruto en la comparativa entre modelo basado en Inercia y el Dinámico. . . . .	202



# Capítulo 1

## Introducción

En esta introducción se va a describir el trabajo realizado en esta tesis doctoral, empezando por la descripción del contexto en el que este tiene lugar, y que se abordará en la sección 1.1.

A continuación, en la sección 1.2, se expondrá la justificación para la elaboración de esta tesis, los diferentes problemas que se tratan de abordar y las soluciones que se aportan para resolverlos. En este trabajo se parten de ciertas hipótesis que se describen en la sección 1.3, necesarias para completar el presente trabajo de investigación, y que ayudarán a cumplir con los objetivos que se enumeran en la sección 1.4.

Para la consecución de los objetivos que se han marcado, se seguirá una metodología ampliamente utilizada y suficientemente madura que se expondrá en la sección 1.5. Finalmente, en la sección 1.6, se describirá la estructura del presente documento.

### 1.1 Contexto

Este trabajo se encuadra en el área de la inteligencia artificial (IA) conocida como computación evolutiva (CE) [175, 176]. La CE pertenece al conjunto de solucionadores de problemas basados en estrategias de prueba y error. Se pueden considerar como solucionadores de amplio espectro, o globales, basados en metaheurísticas, u optimizaciones de tipo estocástico. Para alcanzar sus objetivos, los algoritmos de CE hacen uso de poblaciones de individuos que conforman un

conjunto de soluciones candidatas sobre las que se opera, a diferencia de otros métodos como Simulated Annealing [105, 107], Hill-Climbing [179] o Iterated Local Search [122], que iteran a partir de un determinado punto del dominio de la función de optimización, o espacio de búsqueda, para alcanzar la solución óptima dentro del conjunto de todas las posibles soluciones candidatas a un problema<sup>1</sup>. La CE viene fijada por el tipo de algoritmos que utiliza, conocidos como algoritmos evolutivos (AE) entre los que se incluyen los algoritmos genéticos (AG), la programación genética (PG) y las gramáticas evolutivas (GE)<sup>2</sup>. Los AG fueron introducidos por Holland (1975) [92] y extendidos por Goldberg (1989) [77]. En su trabajo, Holland presenta el marco teórico para la búsqueda de soluciones a problemas mediante el uso de modelos basados en inspiración biológica, a la vez que proporciona una aproximación práctica al uso de los AG. La PG constituye un refinamiento de los AG definido por Kramer (1985) [44] y desarrollado por Koza (1992) [112]. Koza considera que el proceso de resolución de problemas es equivalente a la búsqueda de un determinado programa informático, (dentro del espacio de los posibles programas de ordenador que soporta el lenguaje) o individuo altamente adaptado que lo soluciona.

Con el tiempo, los investigadores han desarrollado una serie de variaciones a partir del algoritmo canónico. Entre ellas, podríamos mencionar las siguientes:

- Cartesian Genetic Programming (CGP) [140].
- Extended Compact Genetic Programming (ECGP) [87].
- Multi Expression Programming (MEP) [166].
- Multiobjective Genetic Programming (MOGP) [166].
- Multitree Genetic Programming (MTGP) [129].
- Linear Genetic Programming (LGP) [31].
- Probabilistic Incremental Program Evolution (PIPE) [182].
- Postfix-GP (Postfix-GP) [46].
- Stack-based GP (SGP) [162].

---

<sup>1</sup> Este tipo de algoritmos normalmente tratan con una única solución candidata que evolucionan.

<sup>2</sup> También conocidas con el término “evolución gramatical”.

- Strongly Typed Genetic Programming (STGP) [141].
- Tree-based Genetic Programming (TGP) [112].

Las gramáticas evolutivas constituyen una técnica de CE que está estrechamente relacionada con la PG. Comparten el objetivo común de buscar y encontrar un programa ejecutable, o un fragmento de programa, que logre un buen valor de aptitud para una función objetivo dada. En las GE, Ryan et al. [181] hacen uso de genomas de longitud variable, representados como vectores enteros, para elaborar definiciones de gramáticas capaces de generar una gran variedad de programas de complejidad arbitraria, mediante la aplicación de operadores genéticos a los vectores de enteros que representan los cromosomas de los individuos.

## 1.2 Definición del problema

Los recientes avances en la tecnología de la información y las comunicaciones han favorecido la automatización progresiva de los sistemas de inversión. La capacidad de los sistemas de inversión algorítmica para procesar datos a alta velocidad (HTF) e identificar regularidades dentro, o entre los mercados, ha llevado a su popularización desde la década de 1990 [36]. Si bien ofrecen ventajas en términos de velocidad de operación; exactitud; reducción de costes; o la ventaja de operar libres de las restricciones que imponen las emociones humanas, por mencionar algunas, su impacto en los mercados financieros en su conjunto sigue siendo controvertido. Los investigadores han identificado tanto elementos positivos, como mayor eficiencia en el proceso de determinación de precios, como negativos, por ejemplo una excesiva volatilidad y mayores costes adversos en la rentabilidad final generada en la selección de la inversión [21,69].

Según Nuti et al. [152], estos sistemas pueden cubrir cualquiera de las etapas específicas del proceso de negocio, incluido el análisis previo a la inversión; la generación de señales de compra/venta; la ejecución de las órdenes; el análisis post-inversión; o todo el ciclo completo de inversión. Este trabajo se centrará en la generación de señales de inversión (compra/venta), y más específicamente en la generación de estrategias de sincronización del mercado.

El rango de alternativas que se pueden usar en este contexto es muy amplio, pero la mayoría de ellas se encuentran dentro de una de las dos categorías siguientes: análisis fundamental y análisis técnico. La primera se centra en el precio justo de un bono, título, obligación, valor o participación, y utiliza los estados financieros como la principal fuente de información de inversión, mientras que la segunda se basa principalmente en las características de los datos de precios y volúmenes de los valores, denominados indicadores técnicos. Estas características, generalmente, se obtienen aplicando fórmulas a las series temporales de precios y volúmenes. Entre los indicadores técnicos, los más conocidos son: los promedios móviles y las líneas de apoyo (soporte o resistencia). Además de estos, otros indicadores clásicos son: la apertura, el cierre, los precios máximo y mínimo, el volumen y la volatilidad [12]. Estos, junto con una serie de operadores y umbrales, se utilizan para definir las reglas técnicas de inversión. Una vez que se especifican estas reglas, los inversores pueden usarlas para tomar decisiones sobre la compra y venta de valores en los mercados financieros.<sup>3</sup>

La cuestión de si es posible encontrar estrategias de inversión rentables para los activos que cotizan en bolsa todavía se encuentra bajo debate. Según Fama [62], y sus seguidores, las acciones se negocian a un valor o precio justo. Si esto se cumpliera, sería imposible vencer al mercado en su conjunto, mediante la selección de los activos negociados o a través de la sincronización del mercado sobre una base ajustada al riesgo. Esta suposición, llamada *Hipótesis de Mercado Eficiente* (EMH, por sus siglas en inglés), tiene tres variantes que lo apoyan: la conocida como “débil”, la “semi-fuerte” y la “fuerte” que difieren entre ellas por la fuerza de la afirmación.

---

<sup>3</sup> Existen probablemente más de 100 indicadores, que de acuerdo a Elder (1993) [59] y Murphy (1999) [144] pueden clasificarse en cuatro categorías. Seguidores de tendencias: los indicadores de esta categoría identifican los principales movimientos de los precios de los activos en un período determinado, ejemplos: media móvil simple, media móvil exponencial, canal Donchian, activador de hilo. Osciladores: los indicadores de esta categoría vigilan las variaciones de precios del activo en un cierto rango de valores, para identificar posibles puntos de retroceso. Ejemplos: oscilador de Chaikin, oscilador de volatilidad de Chaikin, Williams R. Sistemas de banda: los sistemas de banda están constituidos por tres curvas dibujadas alrededor de los precios. Estas curvas se dibujan a partir de una distancia particular de una media móvil. La banda intermedia suele ser un promedio móvil simple, y los intervalos entre las bandas están determinados por la volatilidad del precio. Cuando no hay una tendencia definida, la regla es vender, cuando el precio está por encima de la banda superior, y comprar cuando el precio cae por debajo de la banda inferior. Ejemplos: canal de Keltner, bandas de Bollinger, oscilador de Bollinger. Identificadores de divergencia: estos indicadores se basan en el principio de que toda tendencia pasa por correcciones. Las divergencias ocurren cuando se compara el comportamiento del indicador en relación con el movimiento de precios de un activo. ejemplos: acumulación y distribución, volumen de balance, índice de fuerza relativa. [165].

La EMH considera que en cualquier momento dado se cuenta con toda la información disponible y pertinente. Esta información relativa al mercado es amplia y diversa, presentando varios niveles de conocimiento [63] que pueden ser clasificados, por su profundidad, dentro de alguna de las categorías anteriormente mencionadas:

- **Débil.** Información de los valores, sobre la base de sus precios y volúmenes pasados, hasta el momento considerado.
- **Semifuerte.** Información de precios pasados, u otra información de carácter público que esté disponible, como por ejemplo: el nivel de confianza del consumidor; la tasa de producción industrial; la tasa de paro; el PIB (Producto interior bruto)..., hasta el momento considerado.
- **Fuerte.** Toda la información citada anteriormente, junto a otra de naturaleza privilegiada o interna a la empresa, hasta el momento considerado.

Bajo la conjetura del EMH, ningún sistema debería ser capaz de obtener una rentabilidad superior a una estrategia de tipo comprar y mantener, y menos aún un sistema cuya única fuente de información fuera una incluida dentro de la categoría débil. Sin embargo, intuitivamente, se puede apreciar que los actores en la realidad no disponen de toda la información existente, y que en muchos casos aun disponiendo de ella estos no actúan de forma racional, por lo que se dejan nichos con oportunidades de mercado aprovechables que imposibilitan el cumplimiento de la EMH.

La hipótesis de razonamiento limitado (Limited Reasoning Hypothesis (LRH)) Lakemeyer (1994) [113] es una suposición realista que ayuda completar la imagen de la hipótesis de mercado eficiente, Finger et al. (2004) [66]. Esta hipótesis plantea que, en general, todos los inversores tomarán la mejor decisión posible, pero esta dependerá de la capacidad computacional de la que disponga para procesar la gran cantidad de datos financieros de los que se dispone y que aumentan sin cesar. La hipótesis de razonamiento limitado no contradice el supuesto de racionalidad en las decisiones de la EMH, pero fija su atención en que el inversor es muy posible que no pueda prever todas las opciones posibles, ya sea por la cantidad de información que tiene que considerar o porque carezca de las capacidades de cómputo necesarias para ello [3].

La introducción y uso de las reglas técnicas de inversión entra en confrontación con la creencia de que los mercados financieros siguen un proceso de carácter aleatorio (Random Walk) [64] dirigido por la EMH, lo que se traduce en que cualquier intento de sintetizar una regla de inversión es un trabajo inútil y carente de cualquier valor desde el punto de vista teórico. Esta suposición sostiene que la competencia entre los distintos actores que intervienen en el mismo lleva a una situación de equilibrio en el precio de mercado de un determinado título. Según esta conjetura, el valor del título constituye una buena estimación de su precio teórico o intrínseco. Expresado de otra forma más intuitiva: “Los precios de los títulos de crédito (como las acciones), que se negocian en un mercado financiero eficiente, reflejan toda la información existente y se ajustan total y rápidamente a los nuevos datos que puedan surgir, así como a los cambios en la estructura de mercado, dejando poco espacio a oportunidades de negocio” [62].

Sin embargo, otros autores no están de acuerdo con esta visión [118] y hay toda una industria financiera basada en el argumento de que la inversión activa es rentable.

La inversión activa tiene por objetivo superar la rentabilidad media del mercado, entendida esta como un índice calculado como la media ponderada del conjunto de empresas más importantes, o que están agrupadas en un determinado sector que lo componen. Aunque pudiera parecer que mejorar un índice bursátil, cuya media no tiene en cuenta la situación particular de las empresas, es una tarea fácil, la falta de información que obra en poder de un inversor acerca de las empresas hace que sea muy difícil conseguirlo. Esta estrategia requiere muchos conocimientos e incluso defensores de ella, como Warren Buffett o Howard Marks, reconocen que batir al mercado no es sencillo. “There is no free lunch and beating the market is not easy” [99].<sup>4</sup>

---

<sup>4</sup>De acuerdo con el teorema “No hay almuerzo gratis” (No Free Lunch en inglés (NFL)), para cualquier algoritmo, cualquier rendimiento elevado sobre una clase de problemas tiene exactamente el mismo rendimiento negativo sobre otra clase de problemas [218]. Teniendo en cuenta el teorema de la NFL, Warren Buffett y Howard Marks, probablemente se referían a que es una tarea bastante difícil, sino imposible, predecir qué paradigma de inversión sería el mejor para diferentes índices bursátiles [82]. Aunque es cierto que según el NFL si un algoritmo, pongamos de PG, se comporta mejor que una búsqueda aleatoria en una clase de problemas, se comportará peor que esa misma búsqueda aleatoria en otra clase de problemas, lo cierto es que en general no estamos interesados en la búsqueda del algoritmo perfecto ni en la búsqueda de todas las posibles soluciones. No se debe, por tanto, caer en el error de pensar que no se debe intentar buscar los mejores algoritmos, pues en realidad, lo que buscamos no es el mejor algoritmo general, sino uno que sea capaz de obtener las características que definen las regularidades en el problema de interés que nos ocupa.

La inversión pasiva, por otro lado, se conforma con obtener la rentabilidad del mercado, es decir replicar el índice de referencia elegido. Por ejemplo, un inversor construye una cartera con una serie de activos en porcentajes determinados en los que invierte de manera indefinida. A esta estrategia se la conoce como *Buy and Hold* (B&H), comprar y mantener.

Los detractores de la gestión de inversión pasiva indican que replicar un índice no es tampoco nada fácil, y que en aquellos indicadores que recogen la evolución de muchas acciones (S&P 500, por ejemplo) es extremadamente complejo y costoso lograrlo.

La búsqueda de estrategias de inversión, fiables y rentables, ha impulsado los esfuerzos de investigación durante décadas. Entre los muchos enfoques diferentes, que se pueden usar para diseñar reglas de inversión (como la optimización apoyada en enjambres de partículas, los algoritmos genéticos, las redes neuronales artificiales o los métodos difusos), hay algunos enclavados en ámbito de la computación evolutiva, como las GE y la PG, que son especialmente interesantes debido a tres características clave del proceso: la generación de reglas es automática, las reglas resultantes son interpretables y su estructura es flexible.

Debido a estas tres importantes características, las GE y la PG se han empleado frecuentemente en economía con el fin de modelizar la dinámica de los mercados financieros. No obstante, el número de trabajos en los que se hace uso de técnicas de GE es escaso. Tampoco es posible encontrar investigaciones en las que se compare el uso de las GE y la PG en este ámbito. Una de las posibles causas de esta situación, que menciona Jansen (2011) [102], es que la mayoría de los esfuerzos en este campo se realizan por instituciones financieras privadas, como bancos o agencias de cambio y bolsa, que no están interesadas ni en la publicación de los resultados obtenidos en sus investigaciones ni de los algoritmos que emplean. Los mercados financieros son altamente complejos y dependen de cuantiosos actores que compiten para obtener los mayores beneficios, de ahí que cualquier investigación relacionada con la búsqueda de reglas técnicas de inversión eficientes, que permitan optimizar los resultados haciendo uso de métodos de aprendizaje guiados, sean de gran valor, y entre estos métodos tienen especial importancia las aplicaciones que se apoyan en las técnicas de computación evolutiva [30].

La característica de la interpretabilidad, que proporciona la CE a las reglas de inversión obtenidas, es importante, y la diferencia de otros métodos utiliza-

dos como las redes neuronales profundas [184] (Deep Neural Network (DNN)) que no proporcionan justificación razonada de sus resultados. El mayor desafío frente a estas técnicas no explicativas radica en proporcionar información sobre los procesos que conducen a sus resultados y, por lo tanto, ayudar a aclarar en qué circunstancias se puede confiar en que su desempeño se comporte según lo previsto y cuándo no pueden hacerlo [172]<sup>5</sup>. Es una cuestión que comienza a ser obligatoria y plasmada en diferentes regulaciones. Así por ejemplo, la Unión Europea en su Reglamento General de Protección de Datos [58] (GDPR) disponible<sup>6</sup> en la URL: <https://eur-lex.europa.eu/legal-content/ES/TXT/PDF/?uri=CELEX:32016R0679&from=EN>, establece que: “A efectos del presente Reglamento se entenderá por: «elaboración de perfiles»: toda forma de tratamiento automatizado de datos personales consistente en utilizar datos personales para evaluar determinados aspectos personales de una persona física, en particular para analizar o predecir aspectos relativos al rendimiento profesional, situación económica, salud, preferencias personales, intereses, fiabilidad, comportamiento, ubicación o movimientos de dicha persona física” (Artículo 4, Definiciones, Párrafo 4). De acuerdo con los artículos 13, 14 y 15, cuando los datos personales se recopilan de un sujeto de datos para la toma de decisiones automatizada, el interesado tiene derecho a acceder, y el controlador de datos está obligado a proporcionar “información significativa sobre la lógica involucrada”. El artículo 12 estipula que “la provisión de información a los sujetos de datos debe realizarse de forma concisa, transparente, inteligible y de fácil acceso, utilizando un lenguaje claro y sencillo”.

A lo largo de la historia se ha intentado construir innumerables sistemas de predicción de inversión a partir de series temporales financieras. El problema está lejos de ser resuelto, debido a la dificultad de analizar las series temporales financieras por su alto contenido de ruido. Las series temporales financieras tienen las siguientes características:

- Su estructura, a escalas de tiempo largas, es estructurada y sigue tendencias

---

<sup>5</sup> Los algoritmos de aprendizaje automático dependen de los datos de entrenamiento que se les suministren. Cuando las redes neuronales se enfrentan a un universo completamente observable, y con reglas fijas, como en el ajedrez o el juego de Go, estas pueden aprender de forma autónoma, pero en otros muchos casos se les ha de proporcionar una entrada externa de información. Como consecuencia los algoritmos pueden replicar, o incluso aumentar, comportamientos no deseados del propio entrenamiento: decisiones erróneas, ruido muestral, o sesgos que están presentes en los datos de la muestra. Las redes neuronales proporcionan muy buenas predicciones, pero no son capaces de explicar por qué hizo una predicción en particular. En estas circunstancias puede resultar casi imposible entender las razones o el mecanismo por el que el algoritmo toma una decisión.

<sup>6</sup> Última comprobación de fecha 25/02/2019.



promedio.

- Su estructura a escalas de tiempo corto es aleatoria, presentando muchas fluctuaciones que comportan ruido.

Por lo tanto, una serie temporal financiera dada consta de dos componentes principales: tendencia y fluctuaciones. Las fluctuaciones oscurecen el proceso de modelado de la serie temporal, y hacen extremadamente difícil obtener la tendencia general de la serie. La tarea de la técnica de predicción que se utilice será conseguir extraer la tendencia general de la serie, construyendo un modelo determinista a partir de ella. Estos valores pueden ser un problema serio en la fase de entrenamiento de un AG, ya que pueden generar distorsiones en la evaluación de las reglas de compra y venta de los modelos obtenidos.

El objetivo principal de esta tesis es generar un conjunto de reglas de inversión analizando las series temporales financieras, con la finalidad de extraer, a partir de estas, las características esenciales que definan las reglas obtenidas como: rentables, eficientes, robustas y fáciles de interpretar. Para cumplir con este objetivo, primero es necesario realizar un trabajo exhaustivo de evaluación comparativa de dos de los algoritmos más populares para la generación de reglas de inversión basados en CE que se baraja inicialmente utilizar. Este primer ejercicio de selección de la técnica más apropiada cubrirá, no solo los algoritmos originales basados en GE y PG, sino también el impacto que introduce el agregar dos extensiones a los mismos: funciones definidas automáticamente (en inglés, *Automatically Defined Functions (ADF)*) [111] y mecanismos de control de complejidad de las soluciones o parsimonia. También se estudiará la sensibilidad de los algoritmos a diferentes configuraciones en las que se modificará el tamaño de la población y el conjunto de funciones utilizadas. A partir de los resultados obtenidos, se fijará la técnica a emplear para cumplir con el principal objetivo.

Este primer ejercicio de investigación es esencial, pues desde que Allen y Karjalainen [8] publicaron su artículo seminal sobre la evolución de las reglas de inversión utilizando la programación genética, muchos autores han realizado contribuciones relacionadas basadas en la misma técnica o mediante el uso de alternativas basadas en gramáticas evolutivas. Entre ellos, [61, 96, 120, 121, 133, 146, 147, 187].

La mayoría de estas contribuciones generan reglas de inversión basadas en una

combinación de datos brutos de mercado e indicadores técnicos<sup>7</sup> y, a diferencia de otros enfoques relacionados que utilizan algoritmos genéticos o estrategias evolutivas (EE) [175, 185] para optimizar reglas previamente definidas, tienen la ventaja de crear estructuras mucho más flexibles automáticamente.

Sin embargo, una limitación común es que con frecuencia los algoritmos obtenidos son de naturaleza estática y no tienen en cuenta los cambios estructurales que sufren los mercados. Dado que este fenómeno es prevalente en las series temporales financieras, las reglas de decisión obtenidas en el entrenamiento se derivan, comúnmente, de entornos de mercado que no se mantienen en los períodos de prueba.

El problema de ajustarse a los cambios estructurales es que se debe elegir entre dos extremos opuestos: mantener el mismo modelo a lo largo del tiempo o actualizarlo constantemente. Aunque el segundo enfoque parezca, al menos en principio, más apropiado, existe la posibilidad de que el cambio constante en el modelo aplicado tenga consecuencias indeseables debido a los costes operativos de transacción. El proceso evolutivo de la PG y las GE considera las comisiones a lo largo del período como parte de la función de calidad o aptitud del individuo (*fitness*), y eso lo impulsa a seleccionar reglas que generan un número limitado de señales. Sin embargo, es posible que una actualización constante del modelo de inversión interfiera negativamente con ese mecanismo endógeno de control del número de órdenes de compra y venta. La solución que se busca se puede considerar por tanto multiobjetivo.

En una segunda parte de este trabajo se introdujera un sistema de inversión dinámico basado en técnicas apoyadas en el empleo de GE y en el uso de conjuntos de modelos (*ensembles*) de inversión. El enfoque propuesto combina la posibilidad de modificar dinámicamente el modelo utilizado, como una reacción a los cambios en el mecanismo de generación de precios, junto con un componente de inercia al cambio, que mitiga las consecuencias negativas de un volumen de órdenes de compra/venta excesivo.

La tercera parte del trabajo estará dedicada a una nueva perspectiva de solución que busca, como en el caso anterior, un mecanismo estable que evite excesivos

---

<sup>7</sup> Los indicadores técnicos son fórmulas matemáticas, y estadísticas, que se aplican a las series temporales financieras de precios y volúmenes con la intención de ayudar a tomar decisiones de inversión.

costes de transacción, al tiempo que mantiene la capacidad de modificar el modelo activo, para hacer frente a los cambios derivados del mecanismo de generación de precios del mercado. Dicho enfoque no está basado, como el anterior, en conjuntos de modelos, sino en un sistema dinámico que se aprovecha de los propios mecanismos de histéresis, que forman parte de los modelos obtenidos, para evitar excesivos cambios del mismo.

### **1.3 Hipótesis de trabajo**

En esta tesis, se ha partido de una serie de suposiciones sobre la posibilidad de obtener reglas de inversión eficientes, simples, interpretables, rentables y robustas, a partir de un trabajo de prospección previa en la literatura, sobre técnicas basadas en PG y GE, que ha servido de base y guía para iniciar la investigación.

En una primera hipótesis se ha establecido que es posible desarrollar un sistema de inversión activa que, usando técnicas de CE y análisis técnico, sea capaz de obtener retornos de inversión que batan a una inversión pasiva basada en una estrategia B&H, que se utilizará de referencia. Al sistema se le alimentará con información de mercado de tipo débil, basada en dos conjuntos de series financieras temporales: uno que parte de los precios pasados de cotización bursátil diaria del índice de bolsa Standard & Poor's 500 y otro que contiene datos diarios del retorno sin riesgo, obtenidos del Banco de la Reserva Federal de Atlanta.

Una segunda hipótesis fija que, dentro de las dos técnicas de CE (PG y GE) utilizadas en la literatura, en la búsqueda de reglas de inversión eficientes, son las GE las que: proporcionan los mejores retornos de inversión, las que son más simples, las que son más eficiente desde el punto de vista de coste computacional, las más flexibles y las más robustas en la predecibilidad de los resultados obtenidos.

Una tercera hipótesis reza que es posible mejorar al comportamiento estático de la generación de modelos financieros de inversión, de las GE estándar, mediante una alternativa basada en conjuntos de modelos que combinan la posibilidad de modificar el modelo activo, como una reacción a los cambios en el mecanismo de generación de precios, pero sin la limitación de que este cambio aplicado sea constante.

La cuarta y última hipótesis establece que el uso de un enfoque adaptativo

mediante un mecanismo de selección dinámico, que conmuta entre una regla de inversión activa y una candidata optimizada conforme a los datos de mercado más recientes, proporcionará mejores resultados que la solución de conjuntos de modelos de la hipótesis anterior.

## **1.4 Objetivos**

En esta sección se tratará de fijar los objetivos a cumplir, en los que se enmarca la finalidad de la investigación de esta tesis, y que se enumeran a continuación:

1. Determinar, a través del análisis del estado del arte y la experimentación, qué técnica de computación evolutiva con representación flexible, de entre las más extendidas, resulta más adecuada para generar estrategias de inversión de forma automática.
2. Investigar cómo adaptar la técnica más prometedora, para conseguir que esta sea tolerante al cambio estructural propio de las series financieras. Se examinarán, para ello, diversas líneas de investigación basadas en ventanas deslizantes, sistemas de votación e inercias.
3. Desarrollar algoritmos capaces de generar, de forma automática, estrategias de inversión adaptadas a la incertidumbre provocada por el cambio estructural en series financieras, mediante sistemas dinámicos de cambio de modelos, con el fin de obtener estrategias más robustas al cambio.

## **1.5 Metodología**

Para la consecución de los objetivos de esta investigación se va a seguir la siguiente metodología, cuyos pasos, algunos de los cuales ya han sido tratados en secciones anteriores, se expone a continuación:

1. Definición del problema que se trata de solucionar. Para ello se describirá el objeto que se va a estudiar, es decir, la situación que ha motivado el interés investigador dentro de un contexto que se ha definido. Se busca, de esta

manera, que posteriormente sea fácil comprender la importancia de tratarlo, las limitaciones que se van a encontrar y las contribuciones que los resultados de la investigación puedan aportar a la comunidad científica.

2. Elaboración de una exploración preliminar sobre el estado del arte del tema a investigar. Esto incluye tanto proyectos e investigaciones previas, como trabajos relacionados, así como aportes literarios, científicos y documentales que existen sobre el estado de la cuestión. Todo ello permitirá contar con los antecedentes del tema, con el objetivo de buscar las posibles aportaciones a la solución del problema [89]. El estudio del estado del arte estará dirigido, en primer lugar, a trabajos sobre la generación automática de reglas de inversión usando algoritmos evolutivos. Se hará especial énfasis en aquellas técnicas que cuenten con sistemas de representación flexibles, como la programación genética y las gramáticas evolutivas. En segundo lugar, se analizarán los trabajos existentes que proporcionan soluciones adaptativas dinámicas basadas en marcos temporales, y que cuenten para ello con sistemas de inercia.
3. Realización de una evaluación de las técnicas anteriores, sobre datos históricos, para determinar cuál de ellas es la más prometedora de cara a aplicarla al resto del trabajo.
4. Estudio de las peculiaridades de las series financieras temporales, en relación a su falta de estabilidad en el tiempo.
5. Identificación de estrategias algorítmicas, que permitan adaptar el funcionamiento de la técnica básica seleccionada, para hacerlas capaces de gestionar el cambio estructural de las series financieras.
6. Diseño de propuestas que sean capaces de alcanzar satisfactoriamente los objetivos definidos dadas las hipótesis de trabajo planteadas. La propuesta se describirá con suficiente nivel de detalles, como para permitir la reproducibilidad de los resultados por parte de la comunidad científica.
7. Desarrollo de las diferentes propuestas, que permitan generar estrategias de inversión capaces de ajustarse a condiciones cambiantes del mercado, mediante la implementación del software necesario para materializar la propuesta.
8. Diseño y ejecución de la experimentación.

9. Evaluación de las propuestas contrastando su rendimiento frente a otras estrategias de inversión, utilizando para ello datos reales y aplicando una evaluación estadística exhaustiva de los resultados, con el fin de validar la competitividad de las mismas.
10. Validación de las hipótesis de trabajo y del cumplimiento de los objetivos de investigación. Si algunas de las hipótesis no se sostienen, o si no se alcanzan algunos objetivos, este hecho debe analizarse y las hipótesis subyacentes deben corregirse.
11. Extracción de conclusiones sobre los resultados del trabajo y propuesta de futuras líneas de investigación en este campo.

## 1.6 Estructura del documento

El documento está organizado de la siguiente forma:

- **1 Introducción.** En este capítulo se describe el trabajo realizado en esta tesis doctoral, empezando por la descripción del contexto en el que este tiene lugar, la justificación para la elaboración de esta tesis, los diferentes problemas que se trata de resolver, las soluciones que se aportan para resolverlos, las hipótesis desde las que inicialmente se parte, los objetivos que se pretenden alcanzar y por último la metodología que se seguirá para conseguir los objetivos que se han propuesto.
- **2 Estado del arte.** En este capítulo se hace un repaso de las técnicas relacionadas con la tesis doctoral. Para ello se examinan una serie de trabajos clave, cuyo objetivo es la búsqueda de reglas de inversión automática, y que se clasificarán de acuerdo con el algoritmo subyacente principal, en particular: contribuciones basadas en la programación genética y variantes de esta, contribuciones basadas en gramáticas evolutivas, así como otro conjunto de aportaciones cuyo enfoque es conseguir la robustez de las soluciones y que buscan encontrar la posibilidad de cambiar el modelo dinámicamente, como una reacción a los cambios en el mecanismo de generación de mercado.
- **3 Computación evolutiva.** En este capítulo se hará un repaso de las técnicas de programación genética y gramáticas evolutivas, poniendo especial énfasis

en aquellos aspectos de las mismas que son pertinentes a esta investigación, y que justifican algunas de las decisiones de diseño que se han tomado en la implementación.

- **4 Aspectos generales de la solución.** En este capítulo se tratarán cuestiones comunes a las experimentaciones realizadas, relativas a técnicas de CE utilizadas, como son: el entorno de trabajo; la representación de las reglas de inversión; la descripción de la función de aptitud utilizada; los conjuntos de terminales y no terminales usados en los experimentos o la parametrización común de los algoritmos evolutivos empleados.
- **5 Metodología de evaluación.** En este capítulo se exponen las decisiones tomadas en cuanto a: los tamaños de las muestras en las experimentaciones; el error máximo permitido; el nivel de significación estadística elegido y el protocolo de evaluación de significación estadística utilizado.
- **6 Selección de la técnica apropiada de computación evolutiva.** En este capítulo se describe la primera experimentación realizada en esta tesis, y que consiste en un trabajo de evaluación en el que se comparan los algoritmos de PG y GE, para la generación de reglas de inversión basadas en CE. El objetivo perseguido es mostrar cuál de los dos sistemas tiene un mejor comportamiento en la búsqueda de reglas de inversión, a partir del análisis de series financieras. En el capítulo se presenta el análisis del planteamiento que se compone de: el conjunto de datos utilizado; el diseño experimental; los resultados en términos de rentabilidad: la eficiencia; la robustez y simplicidad de las reglas obtenidos; así como del resumen y las conclusiones.
- **7 Enfoque de conjunto de modelos en inversión algorítmica.** En este capítulo se presenta el análisis experimental, así como los resultados y las conclusiones, del estudio de la posibilidad de cambiar el modelo de inversión estático de las GE tradicionales, mediante técnicas de conjuntos de modelos, como una reacción a los cambios en el mecanismo de generación de precios, pero sin la limitación de que este cambio sea constante.
- **8 Generación dinámica.** En este capítulo se presenta el análisis experimental, los resultados y conclusiones del estudio, en el que se sugiere una solución adaptativa, que involucra un mecanismo de selección dinámico que conmuta entre una regla activa y una regla candidata optimizada, para los datos de

mercado más recientes disponibles, con el objetivo de mejorar los resultados obtenidos en 7.

- **9 Comparativa entre el método de selección dinámica de modelos y el de conjuntos.** En este capítulo se hace una comparación entre la mejor técnica de conjuntos de modelos encontrada y la solución de selección dinámica de modelos.
- **10 Conclusiones y trabajos futuros.** En este capítulo se presentan, de manera desglosada y concreta, las conclusiones y las posibles ampliaciones al trabajo realizado en esta tesis.
- **11 Contribuciones de esta tesis.** En este capítulo se detallan las aportaciones más relevantes realizadas en este trabajo de tesis, así como las contribuciones realizadas en revistas técnicas.
- **Apéndices.** En esta tesis doctoral se ha generado una gran cantidad de experimentación. Debido a la falta de espacio necesario para exponer toda ella en el documento principal, y con el objeto de preservar su legibilidad, sólo aquellas partes más representativas de los planteamientos y su experimentación han sido incluidas en las secciones correspondientes del cuerpo principal, para evitar que la incorporación de todos los resultados obtenidos pudiesen confundir al lector. En esta labor de resumen, se ha buscado identificar y analizar aquellos resultados cuya información es más interesante y relevante, incorporándola en los capítulos dedicados a la experimentación y extracción de las conclusiones, específicamente en los capítulos 6, 7 y 8. El resto de los resultados están disponibles y pueden consultarse en los apéndices correspondientes.
- **Apéndice A.** En este apéndice se recogen las mejores estrategias obtenidas en la configuración base del experimento de selección de técnica de computación evolutiva adecuada. En primer lugar, se muestran, por cada período de entrenamiento, las estrategias ordenadas por técnica, reflejando el valor de aptitud obtenido y la expresión lisp asociada al árbol del individuo. En la segunda parte se presentan los gráficos, en los que se puede ver los momentos temporales en los que el algoritmo específico decide comprar y vender.
- **Apéndices B.** En este apéndice se recoge el protocolo de contraste estadístico



detallado, utilizado para la evaluación de los resultados experimentales de esta tesis.

- **Apéndice C.** En este apéndice se han incluido los resultados de la primera experimentación de selección de la técnica de CE más adecuada para la realización de esta tesis.
- **Glosario.** En esta sección se incluye una recopilación de definiciones, explicaciones de términos o expresiones que versan sobre el tema de la tesis, con una breve descripción de los mismos ordenada de forma alfabética.
- **Acrónimos.** En esta sección se incluye la lista de acrónimos, o abreviaciones más importantes, utilizadas en esta tesis doctoral.
- **Bibliografía.** En esta sección se incluyen las referencias a los autores y publicaciones citadas en este trabajo doctoral.

*Esta página se ha dejado en blanco intencionadamente.*

## Capítulo 2

# Estado del arte

La investigación académica, que trata sobre la generación automática de reglas de inversión mediante el uso de técnicas de CE, como la PG, las GE y sus variantes, tiene una tradición de dos décadas, que comenzó con un primer artículo seminal de Allen y Karjalainen [8] publicado en 1999.

Esta sección consta de dos partes: una primera que se dedicará a presentar una serie de trabajos clave cuyo objetivo es la búsqueda de reglas de inversión automática, y que se clasificarán de acuerdo con el algoritmo subyacente principal; y una segunda que enumera aquellos trabajos cuyo enfoque es conseguir la robustez de las soluciones, que buscan encontrar la posibilidad de cambiar el modelo de inversión dinámicamente, como una reacción a los cambios en el mecanismo de generación de mercado.

### **2.1 Contribuciones que tratan la generación automática de reglas de inversión**

#### **2.1.1 Contribuciones basadas en programación genética**

La mayor parte de la literatura, que trata sobre la generación automática de reglas de inversión, se basa en la versión canónica de la programación genética. El punto de partida de esta revisión sería la contribución aportada por Allen y Karjalainen (1999) [8]. Estos autores intentaron generar, a través de la PG, reglas técnicas de inversión dirigidas al índice Standard & Poor's 500, utilizando como

entrada al algoritmo, datos de precios diarios comprendidos entre los años 1928 y 1995. Su estudio concluyó que las reglas generadas no podían obtener, en los períodos de prueba examinados, excesos de rendimiento de inversión consistentes, que mejoraran la estrategia simple de comprar al principio del período de inversión y vender al final del mismo (B&H), especialmente después de considerar los costes de transacción. Los autores identificaron, no obstante, períodos en los que las reglas señalaban comprar o permanecer en el mercado, en momentos en los que los rendimientos eran positivos y la volatilidad baja, y vender en la situación inversa. Allen y Karjalainen, a diferencia de otros trabajos posteriores, permitían una búsqueda libre en el espacio de los parámetros mediante el uso de constantes generadas aleatoriamente (Ephemeral Random Constants (ERC)), y que aplicaban como coeficientes a los indicadores de medias móviles. El conjunto de funciones no terminales utilizado incluía: operadores relacionales, operadores booleanos y los operadores aritméticos de suma, resta, multiplicación y división.

Tres años más tarde, Setzkorn et al. (2002) [187] trabajaron en una solución que creaba un conjunto de reglas de inversión de mercado que se optimizaban mediante el uso de la PG. El conjunto de reglas estaba formado por relaciones simples entre indicadores técnicos, basados en medias móviles de diferentes lapsos de tiempo, a partir de los cuales se generaban tres tipos de señales: entrar en el mercado, salir de él o permanecer inactivos en la situación en que se encontrasen. Los datos utilizados procedían del índice compuesto S&P 500, y comprendían las sesiones bursátiles entre las fechas 01/02/1990 y el 18/10/2001. Este conjunto de datos se dividió en tres períodos de tiempo: un período de entrenamiento de tres años, uno de validación de cuatro años y uno de prueba de otros cuatro. Los resultados mostraron que las reglas complejas obtenidas eran difíciles de interpretar y adolecían de problemas de sobreentrenamiento, pues se comportaron muy bien en el entrenamiento, pero mal en el período de prueba, mientras que a las reglas simples les fue peor durante el entrenamiento, pero mejor en la fase de prueba, comportándose de forma más robusta ante los cambios de mercado. Los autores consideraron, que la razón más probable del bajo rendimiento del algoritmo, fue el uso exclusivo de medias móviles como indicadores técnicos incluidos dentro del conjunto de funciones terminales de la PG, y pensaron que aumentando su número y variedad probablemente mejorarían los resultados.

El mismo año Thomasy y Sycara (2002) [206] publicaron un documento, en el que evaluaron si la información financiera de los portales de noticias de Internet

(Yahoo y Lycos Finance) tenía un poder predictivo que, utilizado como información de entrada para un algoritmo de PG, pudiera proporcionar rendimientos de inversión superiores a la estrategia de referencia B&H. Para su estudio, emplearon el 10% de los valores con mayor volumen de tráfico de mensajes de Internet, correspondientes al índice Russell 1.000 de las empresas de mayor capitalización de Estados Unidos, generados en el período que iba de enero de 1998 a diciembre del año 2001. La salida del sistema de predicción producía una señal de compra o venta, partiendo de un nivel umbral de tráfico de mensajes prefijado. Si dicho umbral era sobrepasado, se generaba una señal de venta, con el objeto de permanecer fuera del mercado durante un determinado espacio de tiempo. En sus experimentos, la estrategia B&H obtuvo un rendimiento de 126,21% en todo el período de prueba, mientras que el retorno de inversión de la alternativa basada en la PG fue de un 164,36%. En sus conclusiones, los autores afirmaron que el tráfico de mensajes de noticias de Internet contenía información exclusiva sobre el mercado, que no se encontraba en los indicadores numéricos tradicionales (volumen, precio...) y, por tanto, sugirieron que la inclusión de elementos de información procedentes de tráfico de portales de noticias, constituiría una fuente de entrada a tener en cuenta, como parte de un sistema de generación automática de reglas de inversión que utilice la PG.<sup>1</sup>

Becker y Seshadri (2003) [18], siguiendo la línea de Allen y Karjalainen [8], elaboraron una serie de experimentos utilizando la PG, buscando vencer a la estrategia B&H en el índice S&P 500. Su solución tenía en cuenta los costes operativos de las transacciones, pero no incluía los beneficios aportados por los dividendos. Su enfoque difería del aportado por Allen y Karjalainen, en que usaban como entrada al algoritmo datos mensuales, en lugar de datos diarios, con el objeto de reducir los costes operativos asociados a una excesiva negociación. Por otro lado, aumentaron el número de indicadores técnicos derivados, reduciendo al mismo tiempo el número de operadores del conjunto de funciones no terminales, dejando tan sólo los operadores lógicos y los relacionales. Para promover la parsimonia<sup>2</sup>, usaron un factor de penalización de complejidad que aplicaron en el cálculo de la aptitud de

---

<sup>1</sup> En sus experimentos la población estaba constituida por tan sólo 20 individuos, a los que evolucionaban durante 10 generaciones, e imponían una restricción de complejidad máxima de las soluciones generadas de 20 nodos. Además, no aplicaban el operador de cruzamiento y realizaron 30 simulaciones. Justificaron estos valores de parametrización para evitar el sobreentrenamiento.

<sup>2</sup> En este contexto la parsimonia puede definirse como el principio heurístico utilizado para establecer la preferencia por la explicación más simple entre varias alternativas equivalentes.

los individuos. Otro cambio que introdujeron en la función de aptitud consistió en añadir, junto al exceso de rendimiento de inversión obtenido, la ratio de períodos de éxito/fracaso de la regla. Tras el análisis de los resultados, los autores concluyeron que fueron capaces de batir a la estrategia de referencia B&H, demostrando así que la hipótesis EMH no se cumplía. Para ello entrenaron al algoritmo un período de treinta años que iba del 1960 al 1990, y cuyos resultados validaron en el período de doce años de 1991 al 2002. Otra conclusión que aportaron era que, al mismo tiempo que mejoraban a B&H, pudieron reducir de forma importante la complejidad de las reglas encontradas.

En otro estudio contemporáneo al de Becker y Seshadri [18], Neely et al. (2003) [147] usaron la PG, siguiendo la aproximación de Allen y Karjalainen [8], para construir reglas de inversión óptimas, teniendo en cuenta el riesgo ex ante previsto para el índice S&P 500. Para el cálculo del riesgo utilizaron un conjunto de reglas basadas en la ratio de Sharpe [190], la estadísticas  $X^*$  [204] y la medida  $X_{eff}$  [47]. Consideraron diez períodos de estimación superpuestos en la muestra (1929–1935, 1934–1940, 1939–1945... 1974–1980) de datos diarios entre los años 1929 y 1995, para construir diez conjuntos de reglas de inversión ex ante. Con el objetivo de reducir el problema de sobreentrenamiento de los datos, cada período de siete años de la muestra se desglosó en dos fases: una fase de entrenamiento de cinco años y otra de selección de dos. Aquellas reglas que obtuvieron rendimientos en exceso positivos, sobre la estrategia B&H de la primera fase de entrenamiento, se utilizaron para efectuar las pruebas fuera de la muestra con los datos restantes (1936–1995, 1941–1995... 1981–1995), teniendo en cuenta unos costes de transacción de 25 puntos básicos. En sus experimentos, no encontraron evidencia de que las reglas obtenidas superaran significativamente el rendimiento de la estrategia B&H, sobre una base en la que se ajustarán los resultados al riesgo. Concluyeron así, que por un lado sus resultados eran coherentes con la hipótesis EMH, y por otro que para evaluar correctamente las estrategias de inversión, era imprescindible tener en cuenta el ajuste de riesgo.

Al año siguiente, Potvin et al. (2004) [170] propusieron un sistema basado en el uso de la PG, con el objeto de obtener automáticamente reglas de inversión a corto plazo en los mercados de valores. Las reglas de inversión fueron entrenadas y ajustadas utilizando para ello índices de valores individuales, en lugar de emplear algún índice compuesto. En su estudio, se sirvieron de datos de precios de acciones y volúmenes de transacciones pertenecientes a un conjunto de catorce compañías

canadienses, incluidas en el índice bursátil “Toronto Stock Exchange index 300” (TSE 300). El período de estudio empleado comprendía ocho años, que iban desde el 30/06/1992 al 30/06/2000, sumando un total de 2.003 sesiones bursátiles. En sus resultados, encontraron que la PG no fue capaz de vencer a la estrategia B&H. Sin embargo, en un análisis detallado de los resultados, observaron que la solución de PG propuesta lo superó en nueve de los catorce índices contemplados, siendo los otros cinco los responsables del bajo rendimiento alcanzado. Por otro lado, concluyeron que las reglas de inversión producidas no eran productivas en situaciones de inestabilidad o en mercados alcistas, siendo por contra efectivas en mercados estables o bajistas.

Un año más tarde, Fyfe et al. (2005) [71] se sirvieron de la PG para obtener reglas de inversión, que entrenaron tomando como entrada series temporales extensas pertenecientes a los índices bursátiles S&P: índice general S&P 500, y los índices sectoriales S&P Automobile y S&P Banks. El período de datos considerado iba desde el 01/01/1990 al 30/07/1999. Las reglas se evaluaron en términos de sus rendimientos básicos y se ajustaron incluyendo tanto el riesgo de la inversión como los costes de las operaciones. En los resultados obtenidos observaron que, aunque los rendimientos básicos se comportaron muy bien, en comparación con la estrategia B&H, no pudieron superar a esta una vez que se consideraron, tanto el ajuste de riesgo como los costes de operación asociados.

Posteriormente, Navet y Chen (2008) [146] investigaron el uso de la PG en el mercado de bolsa de Nueva York (NYSE), sobre la base de los datos de series temporales de una cartera compuesta por un conjunto de 100 valores del índice. El período de tiempo seleccionado estaba comprendido entre los años 2000 y 2006. Los datos de las sesiones bursátiles se dividieron en tres conjuntos: un primer conjunto de datos de entrenamiento que iba del año 2000 al 2002, un segundo conjunto de datos de validación del año 2003 al 2004 y un tercero de prueba del año 2005 al 2006. Los autores examinaron el rendimiento de las reglas de inversión, elaboradas por la PG, a partir de un esquema de clasificación que distinguía entre valores con niveles de alta y de baja entropía<sup>3</sup>. Los resultados mostraron que la PG sólo superó a una estrategia de inversión de tipo aleatorio, que tomaron como referencia, en dos de las cinco simulaciones efectuadas. En sus conclusiones, los autores señalaron que los datos experimentales obtenidos sugerían, que la predictibilidad de la evolución

<sup>3</sup> Medida de la incertidumbre futura del comportamiento en un proceso dinámico del que el pasado es completamente conocido.

del mercado no era una condición necesaria ni suficiente para obtener rentabilidad. La prueba de predictibilidad sólo era capaz de identificar la existencia de patrones temporales, pero no podía proporcionar información adicional sobre la facilidad o dificultad de descubrir los propios patrones.

En los dos años siguientes, Lohpetch y Corne (2009-2010) [120,121] trataron de identificar una estrategia de entrenamiento, que permitiera la formación de reglas de inversión eficientes, dirigida a superar a la estrategia de referencia B&H de forma consistente. Utilizaron para ello un período de 31 años de datos del índice compuesto S&P 500, comprendido entre los años 1960 y 1991 (en línea con [8, 17, 18]), sobre el que aplicaron dos sistemas diferentes para la elección y evaluación de una regla de inversión obtenida, tras el período de entrenamiento. En la primera estrategia, las reglas obtenidas durante el entrenamiento, que alcanzaron mejores resultados, se evaluaban contra un conjunto de datos de prueba. En la segunda estrategia se añadió un paso más, ya que las reglas obtenidas en entrenamiento se validaban, primero contra un segundo conjunto de datos de validación, y, a partir de ahí, las reglas obtenidas en esta segunda fase se probaban contra un conjunto final de datos de prueba. Los autores concluyeron, que los algoritmos de PG eran muy sensibles a los períodos de datos utilizados, y era claramente mejor usar un conjunto de validación adicional, después del entrenamiento, para elegir la regla a evaluar finalmente. En cuanto a la duración del período de entrenamiento, descubrieron que, en los períodos más cortos, de 18 meses, los modelos eran capaces de batir a la estrategia B&H, mientras que en aquellos más largos, de 24 meses, se revertía la situación anterior siendo el rendimiento de estos peor que B&H.

En el mismo año, Janice How et al. (2010) [96] hicieron uso de algoritmos basados en la PG, siguiendo la misma línea que Allen y Karjalainen [8], para encontrar reglas técnicas de inversión rentables y eficientes. Para ello eligieron cuatro períodos de estimación superpuestos, en la muestra, de siete años (1979–1985, 1984–1990, 1989–1995 y 1994–2000), cuyos datos diarios se utilizaron como entrada del algoritmo de PG. Para evitar caer en sobreentrenamiento, cada período dentro de la muestra se dividió a su vez en dos períodos, un primer período de entrenamiento de cinco años y segundo período de selección de dos años. Las reglas así generadas fueron a continuación probadas, con datos fuera de la muestra, usando el resto del conjunto de datos disponibles como período de prueba, que siempre se iniciaba en el año inmediatamente posterior al correspondiente período de selección: 1986, 1991, 1996 y 2001, y terminaba el 31 de diciembre del año 2005.



El proceso de entrenamiento, selección y evaluación era similar al realizado por Neely [147]. En sus resultados, de la misma forma que Brock [33], Allen y Karjalainen [8], y Potvin et al. [170], encontraron que las reglas obtenidas tendían a aconsejar al inversor que permaneciera en el mercado durante los períodos de baja volatilidad y rendimientos positivos, y que se mantuviera fuera del mercado durante los períodos de alta volatilidad y rendimientos negativos.

Del mismo modo que Potvin et al. [170], Esfahanipour et al. (2011) [61] se apoyaron en técnicas de PG para generar automáticamente reglas técnicas de inversión eficientes, considerando la importancia que introduce el factor riesgo en la toma de decisiones del inversor, y el efecto que introducen los costes de transacción, los dividendos y los splits, en los resultados. Los autores introdujeron una modificación en la estructura de generación de recomendaciones, de las reglas de inversión, ampliando el número de señales de dos (comprar/vender) a tres: comprar, vender y mantener. Para el cálculo de ajuste del factor riesgo utilizaron la relación condicional de Sharpe [190], debido a que contiene menores supuestos y proporciona más precisión, en la medición del riesgo, que otros sistemas. La aplicación del modelo de PG utilizado se llevó a cabo de dos maneras: una en la que se ajustó el resultado en función del riesgo, y otra en la que este no fue considerado. Los autores implementaron su solución teniendo en cuenta los índices de acciones individuales, de una cartera formada por un conjunto de diez compañías iraníes pertenecientes al mercado de la bolsa de Teherán (Tehran Stock Exchange (TSE)). Utilizaron un período de entrenamiento de tres años que iba del 22/8/2004 al 21/8/2007, y otro de prueba del 22/8/2007 al 21/8/2008. En la solución de la PG, en la que se aplicó el modelo de riesgo no ajustado, los resultados obtenidos fueron similares a los publicados por Potvin et al. [170], en el sentido en que sus reglas de inversión superaron a la estrategia B&H en mercados estables y bajistas y no fueron rentables en mercados extremadamente alcistas. Por el contrario, las reglas del modelo de PG ajustadas al riesgo, superaron a la estrategia B&H en todas las condiciones del mercado, tanto en aquellas en las que el mercado era alcista, como en las que el mercado se comportaba a la baja. Llegaron a la conclusión de que su modelo de PG consiguió producir reglas de inversión rentables para todos tipos de inversores, tanto para los que asumen el riesgo como para quienes intentan evitarlo.

En su tesis, Jansen (2011) [102] eligió una muestra más reciente del índice de la bolsa de valores de Alemania (Deutscher Aktienindex (DAX)) y del índice bursátil chino Hang Seng (HSI), que usó para generar estrategias de inversión eficientes

usando técnicas de PG. El conjunto de datos comprendía el período de sesiones bursátiles que iba desde el año 1997 al 2007. En la experimentación estableció que los conjuntos de datos de entrenamiento tuvieran un tamaño de tres y cinco años, que combinó con conjuntos de datos de prueba de uno y tres años respectivamente. El autor realizó algunos experimentos con períodos de entrenamiento más largos, de diez años, en donde las reglas obtenidas fueron validadas con un período de prueba de un año. A pesar del hecho de que el algoritmo de PG se entrenó con un conjunto de datos que incluía un ciclo económico completo, los resultados obtenidos fueron negativos. En sus conclusiones, afirmó que las reglas de inversión generadas por la PG, generalmente produjeron resultados negativos en términos de rendimiento y, lo que es más significativo, en la ratio de riesgo de Sortino [178]. El autor entendió que los resultados obtenidos podían implicar que la PG no era capaz de batir a la estrategia de referencia B&H. Este fue especialmente el caso, una vez que empleo un coste de transacción más realista de un 0,25 %, en lugar del 0,1 % que utilizó como punto de partida. Concluye así que se cumple con el EMH, ya que las reglas más rentables que se consiguen son o bien negativas, o básicamente siguen una estrategia de B&H. Aunque en un análisis posterior expuso que podrían existir algunas situaciones, en las fuese posible localizar ineficiencias en un mercado de tipo altamente maduro y líquido, que se podían aprovechar para mejorar la rentabilidad obtenida.

Tres años más tarde, Gabrielsson et al. (2014) [72] exploraron la viabilidad de desarrollar estrategias de inversión, apoyándose en las GE, que señalizaran la entrada y salida del mercado de forma transparente, y dirigidas al mercado de futuros del índice E-mini S&P 500, en un entorno de inversión de alta frecuencia de negociación (en inglés: High Trading Frequency (HTF)). Compararon el rendimiento de los modelos que incorporan el riesgo en sus cálculos, con otros modelos que no lo hicieron. En sus conclusiones, afirmaron que los resultados empíricos obtenidos sugerían que se pueden obtener estrategias de inversión transparentes y rentables, que evitan al riesgo, en el entorno del índice E-mini S&P 500, combinando el uso de las GE junto con el empleo de indicadores técnicos.

En el año 2015, Luengo et al. (2015) [124] estudiaron el comportamiento de la aplicación de un conjunto de reglas de inversión simples optimizadas mediante técnicas de PG, buscando encontrar, mediante ellas, un método eficiente para generar señales de entrada y salida en el mercado de valores español. El ámbito de prueba se aplicó a tres escenarios diferentes del estado del mercado: en mercados

cuya tendencia es alcista, en mercados bajistas y en mercados laterales. Para evaluar el rendimiento de su método, realizaron una comparación con otras estrategias de inversión tradicionales, como la de B&H o la del cruce de promedios móviles simples. En sus resultados, encontraron que el comportamiento global del mercado tuvo una gran influencia en los resultados de cada método, y descubrieron que las estrategias basadas en la PG tuvieron mejores resultados en mercados de tipo lateral.

En ese mismo año, Gypteau et al. (2015) [85] usaron una escala de tiempo intrínseca, basada en cambios direccionales, que combinaron con la PG para encontrar una estrategia de inversión óptima, con la que pronosticar los futuros movimientos de precios de los mercados financieros. Su hipótesis de trabajo partía de la observación de que la mayoría de las herramientas de pronóstico utilizaban una escala de tiempo física, para estudiar las fluctuaciones de los precios de los mercados financieros, lo que hacía que el flujo de tiempo físico fuera discontinuo, y por lo tanto, el uso de una escala de tiempo física podía exponer a las empresas a riesgos, debido al desconocimiento de algunas actividades significativas. Los autores evaluaron la eficiencia y robustez de su solución, como herramienta de pronóstico, a través de una serie de experimentos y concluyeron que, con su enfoque, eran capaces de obtener información valiosa sobre el rendimiento de la predicción. Los resultados experimentales avalaban que este nuevo marco de trabajo podía generar estrategias de inversión nuevas y rentables.

Hongguang et al. (2015) [95] usaron series temporales financieras intradía, como entrada a un sistema de PG, con el objetivo de explotar completamente la ventaja que proporcionan los métodos de análisis técnicos, como sistemas de predicción de inversión a corto plazo. Pensaron que la utilización de datos intradía, para entrenar las reglas de inversión, podría evitar puntos de salto en los datos diarios o mensuales, ya que se supone que la información relevante se digiere completamente durante los períodos de cierre del mercado. Por un lado, sus resultados mostraron que las reglas de inversión generadas por la PG fueron capaces de acumular ganancias de negociación dentro de un período determinado (los datos de la sesión de la mañana tenían más probabilidades de ser explotados, de manera rentable, en la sesión de la tarde del mismo día de negociación), sin embargo las estrategias entrenadas con datos de varias sesiones podían llevar a pérdidas severas, en varios de los siguientes días de negociación. En su opinión, esto indicaba que la tendencia del mercado presentaba una inclinación a modificarse tras varios días

de negociación, mientras que se mantenía igual cuando la negociación tenía lugar dentro de un mismo día. Como datos de entrada al algoritmo utilizaron 21 días de datos de futuros, pertenecientes al índice del mercado chino de futuros Shanghai Shenzhen 300, también conocido como Hushen 300, que cubrían el período de sesiones, que iba del 29/09/2014 al 03/11/2014.

Dos años más tarde, Pimienta et al. (2017) [165] propusieron un sistema computacional que combinaba un método de negociación convencional basado en el análisis técnico, algoritmos de PG y una optimización de tipo multiobjetivo, para generar un sistema de inversión automatizado. Es interesante observar, que presentaron un mecanismo para la detección y eliminación automática de valores atípicos, a fin de minimizar las distorsiones en la evaluación de las reglas candidatas de compra y venta. El sistema de inversión automatizado propuesto se aplicó a seis títulos bursátiles (Banco do Brasil SA (BBAS3), ISHS IBOVESPA FDO índice ETF (BOVA11), Companhia Energetica de Minas Gers, CEMIG Preference Shares (CMIG4), IEmbraer SA. (EMBR3), Gerdau SA Preference Shares (GGBR4) y el indicador de resistencia de precios (VALE5) de la Bolsa de Valores do Estado de São Paulo (BOVESPA), con una ventana de prueba de 514 días hábiles de negociación, tomada entre las fechas del 02/05/2013 al 02/02/2015. A partir de los resultados mostraron, que su sistema, pudo obtener rendimientos financieros considerablemente por encima del precio de variación de las acciones en el mismo período, superando a otras dos estrategias de inversión automatizadas con las que se comparó B&H, TA-10, y TA-36<sup>4</sup>. Además, pudo obtener ganancias significativas, incluso en situaciones de fuerte depreciación del activo.

Un año más tarde, Yang et al. (2017) [221] presentaron un marco de trabajo para desarrollar estrategias de negociación, motivado por los hallazgos empíricos que mostraban que las noticias y los mensajes de Twitter, en las redes sociales (tweets), exhibían un poder predictivo persistente en el movimiento del mercado financiero. Los autores tuvieron en cuenta los estudios que evidenciaban, que los tweets eran más rápidos que las noticias al revelar nueva información de mercado, mientras que las noticias se consideraban, en general, una fuente de información más fiable que los tweets, para proponer una estrategia de inversión basada en la fortaleza de correlar los comentarios entre las noticias y los tweets. El sistema hacía uso de métodos de optimización, basados en PG, para obtener las señales de pre-

---

<sup>4</sup> Ambas estrategias TA-10 y TA-36 hacen uso de ventanas deslizantes y sistemas de conjuntos.

dicción de compra/venta. La intuición clave detrás de su enfoque, fundamentado en la fuerza de retroalimentación, era que el impulso conjunto de las dos series de sentimientos, conducía a señales significativas del mercado, que se podían explotar para generar mayores retornos de inversión. En su estudio, eligieron el ETF S&P 500 como una representación amplia del rendimiento del mercado de los EE.UU. y obtuvieron el rendimiento histórico diario de estos índices, a través de un terminal de Bloomberg, tomando como entrada los datos que iban desde el 31 de julio del 2012 hasta el 30 de enero del 2015. Encontraron que, durante el período analizado, el indicador de sentimiento basado en un enfoque de PG arrojó rendimientos de mercado superiores, con una reducción máxima de promedio mensual baja. Cuando compararon la ratio de Sterling [108], y otras medidas de riesgo, las estrategias propuestas, basadas en los indicadores de sentimiento, fueron superiores a las estrategias basadas en indicadores técnicos y a la estrategia tradicional de B&H. En sus conclusiones, sugirieron que los resultados avalaban que el uso de un indicador de retroalimentación de opiniones brindaba un óptimo apoyo para controlar las pérdidas, con una reducción máxima más baja.

### 2.1.2 Contribuciones basadas en variantes de la programación genética

A pesar de que la mayor parte de la literatura se centra en el estudio de la versión estándar de la PG, varios autores han basado sus investigaciones en variaciones de esta que amplían sus capacidades o que mitigan algunas de sus limitaciones.

Mousavi et al. (2014) [142], por ejemplo, emplearon la PG para desarrollar un sistema dinámico de negociación de cartera, para capturar la dinámica de los precios del mercado de valores a través del tiempo. El enfoque propuesto tenía una visión integrada compuesta de múltiples valores y generaba una base de reglas, que descansaba en el uso de indicadores técnicos, para la negociación dinámica de la cartera de valores. En su investigación, desarrollaron un bosque de PG de múltiples árboles (Multitree Forest) para extender la estructura de la PG, con el objeto de extraer múltiples reglas de inversión, a partir de datos financieros históricos. Además, su propuesta tenía en cuenta los costes de transacción de la negociación, pues desempeñaban un papel importante en la rentabilidad de un sistema dinámico de negociación de cartera. Este modelo se utilizó para desarrollar sistemas dinámicos de inversión. Los datos utilizados fueron obtenidos a partir de una muestra

de los mercados de bolsa iraníes y canadienses. Sus resultados mostraron, que la solución propuesta, superó significativamente a otros modelos tradicionales de selección dinámica y estática de cartera, en términos de rendimiento de la cartera y de rendimiento ajustado al riesgo.

Dabhi y Chaudhary (2015) [45] propusieron un sistema que combinaba el uso de ondículas (Wavelet) y Postfix-GP, un sistema de programación genética basado en notaciones postfix, para la predicción de series temporales financieras. El uso de la transformada discreta de ondículas se utilizó para suavizar las series temporales financieras, al separar las fluctuaciones de la tendencia general de la serie. Los autores realizaron los experimentos sobre cuatro series temporales financieras, dos series de precios de las acciones de Intel y Microsoft y dos índices generales de la bolsa de valores NASDAQ Composite y S&P CNX Nifty. Los períodos utilizados iban del 12/09/2007 al 11/11/2010 para las compañías Intel y Microsoft, del 01/03/2007 al 22/03/2011 para el índice NASDAQ, y del 01/03/2007 al 08/04/2001 para el índice S&P CNX. El resultado de este trabajo se comparó con los resultados obtenidos por un método de PG tradicional. Las pruebas indicaron que el trabajo propuesto fue mejor en los dos índices de bolsa NASDAQ y S&P CNX, pero no se llegó a un resultado concluyente en la comparación realizada con los índices de las empresas Intel y Microsoft.

Más recientemente, dos artículos presentaron soluciones basadas en el uso de la STGP. Este enfoque impone restricciones en los tipos de datos y utiliza funciones genéricas, así como tipos de datos, que resuelven algunos problemas relacionados con los requisitos de cierre de la versión canónica de la PG.

Manahov et al. (2015) [133] desarrollaron un sistema de predicción, dirigido a obtener modelos de inversión rentables en el mercado de valores. El sistema trabajaba sobre diversos instrumentos financieros y de carteras de valores, utilizando una forma adaptativa especial de la programación genética, la STGP. El algoritmo de inversión, basado en la STGP, realizaba predicciones de rendimiento con un día de anticipación, utilizando para ello diferentes grupos de inversores artificiales, o agentes, de diferentes tamaños y con diferentes niveles de inteligencia. El período de la muestra, seleccionado para el experimento, consistió en 11.405 datos de sesión diarios, de más de 45 años, pertenecientes al índice S&P 500, que iban del 24/05/1962 al 14/09/2007. También se incluyeron las cotizaciones diarias de las acciones de las empresas IBM y General Electric. El período fuera de la muestra

incluyó 1.515 datos de sesión diarios, de una duración de cinco años, que iba del 17/09/2007 al 17/09/2013. En sus conclusiones señalaron que encontraron poco apoyo para la hipótesis del inversor marginal [68],<sup>5</sup> pero alguna evidencia a favor de la Hipótesis de Hayek. [88]<sup>6</sup>

Un año más tarde, Agapitos et al. (2016) [2] utilizaron una variante de la programación genética, con control fuerte de tipos (STGP), para modelar sistemas de predicción de inversión. El programa generado, con capacidad de memoria, fue comparado con una versión estándar de la PG en varias tareas financieras de modelado de series temporales. Basaron sus experimentos en datos procedentes del índice bursátil S&P 500 EUR/USD y del Nikkei, y utilizaron, en el algoritmo, un conjunto de funciones no terminales complejas que incluían: funciones matemáticas, funciones lógicas booleanas, funciones de acceso a memoria, operadores relacionales, operadores condicionales e instrucciones condicionales. El conjunto de elementos terminales estaba compuesto de indicadores técnicos y ERCs. En sus conclusiones afirmaron, que los programas que usaban memoria generalizan mejor que sus homólogos estándar de la PG, en la mayoría de los conjuntos de datos del dominio del problema analizados.

### 2.1.3 Contribuciones basadas en gramáticas evolutivas

El número de artículos publicados que aplican las técnicas de las GE, en este dominio, es más limitado. Esto no es sorprendente, ya que el propio cuerpo de literatura, sobre la PG, en general también es significativamente más numeroso que el dedicado a las GE.

El primer trabajo, del que el autor de esta tesis tenga conocimiento, en el que se hizo uso de las GE, para construir sistemas de predicción financiera corresponde al de Brabazon y O'Neill (2004) [29]. Los autores exploraron la posibilidad de utilizar, como instrumento de trabajo, técnicas basadas en las GE, para construir un sistema de predicción, que generase reglas de inversión destinadas a superar la estrategia

---

<sup>5</sup> La hipótesis del inversor marginal (Marginal Trader Hypothesis (MTH)) explica la eficiencia del mercado a través de las acciones de los grandes inversores, que capitalizan las valoraciones incorrectas en el mercado, con el fin de maximizar sus propias ganancias y corregir el precio en el proceso.

<sup>6</sup> Friedrich Hayek afirmó, en lo que ha llegado a ser llamada la hipótesis de Hayek, que a pesar de las limitaciones que los inversores tienen del conocimiento del entorno, y de la situación de otros inversores, los mercados pueden funcionar correctamente.

de referencia B&H en el mercado monetario. Sus resultados superaron el índice de referencia B&H en cinco de los seis conjuntos de prueba, utilizando una métrica que penalizaba la aptitud del individuo en función del riesgo comercial de la inversión, un conjunto de indicadores técnicos reducido como funciones no terminales, y una cantidad relativamente baja de datos de divisas del mercado de Londres, del período de cinco años comprendido entre el 23/10/1992 y el 13/10/1997.

El mismo año, Dempsey et al. (2004) [52] usaron una metodología de programación automática evolutiva, basada en las GE, para producir reglas técnicas de inversión aplicables a los índices S&P 500 americano y el Nikkei 225 japonés. Los autores analizaron dos soluciones: una que utilizaba una sola población de reglas, que se adaptaba a lo largo de las series temporales de datos financieros y otra mediante la cual se creaba una nueva población, para cada ciclo generacional del algoritmo evolutivo. En sus resultados experimentales obtuvieron retornos positivos en los períodos de tiempo contemplados, con claras ventajas en el caso de la población adaptativa de reglas. Sin embargo, cuando analizaron por separado cada una de las bolsas, encontraron que en el índice S&P 500, excepto en cinco de las treinta simulaciones efectuadas, había muy pocas oportunidades para superar el índice de referencia B&H. Por otro lado, al analizar el índice Nikkei 225, observaron que las GE generaron rendimientos, por retornos de inversión, con una mejora promedio del 74 % sobre el índice de referencia.

En su artículo, Contreras et al. (2013) [41] presentaron un sistema de inversión financiera basado en las GE. Los autores probaron su rendimiento, comparando sus resultados contra la rentabilidad histórica de un conjunto de empresas del mercado español, utilizando para ello datos de sesiones correspondientes al año 2012. Además de eso, contrastaron el sistema propuesto, que descansaba en el uso de las GE, con un trabajo previo [40] donde los autores aplicaron un enfoque basado en AG. En sus resultados, constataron que el sistema de inversión implementado con las GE, obtuvo un rendimiento del 14 %, mientras que la alternativa basada en los AG obtuvo una pérdida de aproximadamente el 20 %. El análisis, con un conjunto extendido de nueve empresas españolas seleccionadas, mostró que la rentabilidad general obtenida, era mayor que la estrategia de referencia B&H.

Schmidbauer et al. (2014) [183] construyeron, y probaron, un sistema marco para la selección de reglas de inversión, que frenaba el sesgo en la inferencia estadística (data-snooping) en la evaluación del rendimiento. El núcleo de su enfoque



residía en el concepto de robustez a priori, por el que una regla de inversión, que funcionase bien en la serie temporal original de precios, también debería funcionar bien cuando se expusiese a un escenario alternativo, exhibiendo características similares, pero sin que estas fuesen idénticas a las series de tiempo originales. Los autores desplegaron una herramienta de computación evolutiva, basada en una gramática guiada sobre la PG para el proceso de selección, y un criterio de cálculo de aptitud que fijaba varios objetivos que involucraban tanto a la serie temporal original, como a la modificada. El método se probó en el mercado de divisas, en operaciones de cambio monetario realizadas en euros/dólares, con datos intradía, que iban de febrero a junio del año 2011. Sus hallazgos sugirieron que el criterio de robustez a priori daba menos resultados espurios, y eso evitaba el sobreajuste obtenido a partir de la muestra de entrenamiento. En sus resultados, si bien encontraron evidencias de que la ganancia en la muestra de prueba podía incrementarse, esta aún no podía alcanzar un valor de rentabilidad.

Finalmente, vale la pena mencionar, la contribución de Berutich et al. (2016) [20], que presentaron un sólido enfoque de PG para descubrir reglas de inversión rentables. Las reglas obtenidas se utilizaron, posteriormente, para administrar una cartera de valores pertenecientes al índice de la bolsa de valores española IBEX35. En su solución, los autores exploraron un método de muestreo aleatorio de aptitud en la PG (en inglés: Random Sampled Fitness Genetic Programming (RSFGP)). El algoritmo central de este sistema era básicamente como el estándar de la PG, pero en lugar de calcular la aptitud del individuo sobre el conjunto completo de datos, estos se dividían en un conjunto de segmentos seleccionados al azar, con el objetivo de aumentar la robustez de las predicciones. En sus conclusiones, explicaron que el sistema basado en RSFGP era capaz de tratar con diferentes tipos de mercados, y lograron un rendimiento de la cartera de inversión española del 31,81 %, para el período de prueba comprendido entre el año 2009 y el 2013, en comparación con el 2,67 % del rendimiento obtenido en el índice IBEX35.

La tabla 2.1, resume la información más importante de esta sección, ya que clasifica las piezas de investigación mencionadas según el algoritmo central utilizado, PG o GE. También muestra si los autores utilizaron algún mecanismo de control de complejidad (parsimonia), distinto a la aplicación de una mera restricción en el tamaño de los árboles de los individuos, ya sea en términos de complejidad por número de nodos, o de profundidad alcanzada. Como se mencionó anteriormente, es evidente que el algoritmo central más popular es el de la PG, que se usa en 23 de

**Tabla 2.1:** Trabajos relacionados. Estrategias y técnicas de CE utilizadas.

Trabajos relacionados	Año	GE	PG	Parsimonia
Allen and Karjalainen	(1999)		X	
Setzkorn et al.	(2002)		X	
Thomasy y Sycara	(2002)		X	
Becker and Seshadri	(2003)		X	X
Neely	(2003)		X	
Brabazon et al.	(2004)	X		
Fyfe et al.	(2005)		X	
Potvin et al.	(2006)		X	
Dempsey et al.	(2006)	X		
Navet y Chen	(2008)		X	
Lohpetch et al.	(2009)		X	X
Jensen	(2010)		X	
How	(2010)		X	
Esfahanipour et al.	(2011)		X	
Contreras et al.	(2013)	X		
Schmidbauer	(2014)	X		
Gabrielsson et al.	(2014)		X	
Mousavi et al.	(2014)		X	
Dabhi y Chaudhary	(2015)		X	
Gypteau et al.	(2015)		X	
Luengo et al.	(2015)		X	
Manahov	(2015)		X	
Hongguang et al.	(2015)		X	
Agapitos et al.	(2016)		X	
Berutich et al.	(2016)		X	
Pimienta et al.	(2017)		X	
Yang et al.	(2017)		X	

(a) Ordenados por fecha y con expresión de si usan PG o GE, con o sin parsimonia.

los 27 trabajos considerados, frente a los cuatro de las GE. El uso de estrategias de promoción de la parsimonia, sólo se menciona en dos documentos de la muestra, y es notable que nadie haya explorado la conveniencia de agregar ADF como extensiones del algoritmo central. Todas estas posibilidades serán evaluadas en este trabajo de tesis doctoral.

## 2.2 Contribuciones basadas en sistemas dinámicos

Si consideramos soluciones adaptativas, que buscan mejorar la robustez de los modelos predictivos, incluso si abrimos las posibilidades de las GE a la CE en general, el número de enfoques relevantes encontrado es bastante limitado. Entre las referencias más relacionadas con la sugerida en este estudio, podemos destacar las que se relacionan a continuación.

A principios de siglo, Dempster et al. (2001) [53] realizaron un trabajo en el que trataron de emular el comportamiento de los inversores, mediante el desarrollo de un sistema automático de inversión financiera. El sistema perseguía la generación de reglas de negocio, mediante técnicas de CE, sustentadas en combinaciones de diferentes indicadores técnicos, que aplicaban con distintas frecuencias y retrasos. Los autores comenzaron el análisis de la solución partiendo de la evidencia de que los indicadores de análisis técnico eran ampliamente utilizados por los inversores, que buscaban predecir los niveles de precios futuros y mejorar la rentabilidad de la inversión, en los mercados financieros y de productos básicos. Su hipótesis consistía en que se podía demostrar que existía una serie de reglas de inversión muy populares, basadas en indicadores técnicos, que eran capaces de generar pérdidas cuando estas eran aplicadas individualmente de manera sistemática. Sus observaciones repararon en el hecho de que los inversores solían utilizar combinaciones de indicadores técnicos, y no sólo uno, que eran tomados dentro de una amplia gama de ellos. Por otro lado, también tuvieron en cuenta, que los inversores que tenían más éxito tendían a adaptarse a las condiciones cambiantes del mercado, al “eliminar” las reglas de inversión de su repertorio tan pronto como estas generaban pérdidas, o cuando, por otra parte, encontraban reglas que resultaban ser más rentables. Su solución partía de un portafolio inicial de reglas de uso común, que se seleccionaba mediante un algoritmo genético aplicado a una serie de indicadores, calculados a partir de un conjunto de datos de entrenamiento de moneda extran-

jera. La serie temporal financiera utilizada tenía un período de tres años, que iban del año 1994 al 1997, y estaba compuesta de datos agregados de varias frecuencias intradía. El algoritmo genético se usaba, posteriormente, a intervalos regulares en los datos fuera de la muestra, para proporcionar nuevas reglas. A continuación, se utilizaba un sistema de retroalimentación, para reequilibrar la cartera de reglas, creando así dos niveles de adaptabilidad. A pesar de que los indicadores individuales generalmente generaban pérdidas, durante el período de datos de prueba, la mejor regla encontrada por el sistema desarrollado fue modesta, pero significativamente rentable, en presencia de costes de transacción realistas.

Grosan et al. (2006) [83] usaron dos técnicas de programación genética MEP y LGP, para construir un conjunto de predicción de dos índices bursátiles diferentes: el índice Nasdaq-100 (perteneciente al Nasdaq Stock Market (NSM)) y las acciones de S&P CNX NIFTY. En el trabajo se comparó el rendimiento de la solución, con una una red neuronal artificial entrenada utilizando el algoritmo de Levenberg-Marquardt [23], el modelo neuro-difuso de Takagi-Sugeno (Takagi-Sugeno fuzzy inference system (NF)) [1, 101, 203], y los algoritmos de programación genética MEP y LGP. Para evolucionar los algoritmos de PG aplicaron un algoritmo de optimización evolutivo multiobjetivo (MOEA) conocido como "*Algoritmo Genético de Clasificación no Dominado II*" (Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization, en inglés (NSGAI)) [51]. Los resultados empíricos revelaron que las técnicas de conjunto utilizadas, en programación genética, constituían un método muy prometedor para uso en sistemas de predicción de tendencias bursátiles. Los autores concluyeron, que los resultados obtenidos por el método de conjunto fueron mejores que los alcanzados por cada una de las técnicas de PG aplicadas individualmente.

Una década más tarde Wilson et al. (2011) [217] usaron técnicas de programación genético lineal (LGP), con datos de sesiones sobre múltiples marcos de tiempo intradía, para desarrollar un sistema que generaba diversos modelos de inversión financiera. Los modelos producían señales de predicción de mercado, en las que se señalaba comprar o vender como respuesta a la estrategia de inversión que seguía el modelo. Los autores establecieron dos sistemas de decisión para determinar la acción del modelo de inversión, en donde establecía la estrategia de órdenes de entrada o salida finales del mercado, que era después aplicada a todos los marcos de tiempo. Uno de ellos se basó en el voto mayoritario, y la señal de compra o venta se generaba si esa señal representaba a la mitad, o más, de todas las señales indi-

viduales de cada escala de tiempo. El otro modelo se sostenía por la proximidad temporal a la decisión de compra. En sus conclusiones encontraron que el mecanismo de decisión de proximidad temporal era más restrictivo, y negociaba con una menor frecuencia que la alternativa basada en una decisión mayoritaria. Sin embargo, al realizar el análisis de rendimientos, encontraron que no funcionaba tan bien durante el período de prueba. La decisión mayoritaria que incluía un mayor número de votantes, por otro lado, fue ligeramente menos reactiva a los cambios en la tendencia de los precios. Dentro del propio modelo de decisión mayoritaria se encontró que el aumento en el número de votantes, a través de más marcos de tiempo, se comportaba mejor que las combinaciones que usaban marcos de tiempo más cortos. Los autores pensaron, que la razón de este comportamiento era debida a que un mayor número de votantes alentaba a permanecer dentro del mercado, y reducía el número de operaciones de compra o venta totales.

Dos años más tarde Shangkun Deng et al. (2013) [54] utilizaron técnicas basadas en AG, para generar reglas de inversión financieras en el mercado de divisas. Las reglas se apoyaban en el uso de un sólo indicador técnico denominado índice de fuerza relativa (RSI). Los autores incorporaron, como entrada a un AG, datos organizados en tres marcos de tiempo, a partir de los cuales se pretendía extraer las características de una serie temporal financiera. Como objetivo del estudio, utilizaron el par de divisas negociables con mayor volumen de contratación: el euro EUR y el dolar americano USD, siendo el horizonte de negociación temporal base fijado de una hora. Los tres marcos de tiempo, con el que los autores alimentaron al sistema, se componían de una señal combinada mediante los citados tres marcos de tiempo: uno con un período de tiempo relativamente largo de dos horas, uno de un período de tiempo corto de treinta minutos, y un tercero con un período de tiempo objetivo base de una hora. El conjunto de datos cubrió el lapso de tiempo comprendido desde el 3 de enero del 2011 al 30 de diciembre del 2011, sumando un total de 6.178 observaciones de datos de una hora. En sus resultados experimentales concluyeron que la señal combinada, a partir de múltiples marcos de tiempo, mejoraba el rendimiento de las operaciones de inversión.

Siguiendo una estrategia diferente, que combinaba un conjunto de reglas múltiples basadas en indicadores técnicos, Jayanthi et al. (2014) [145] sugirieron un nuevo enfoque para obtener una estrategia de inversión dirigida a los mercados de valores. Sus resultados experimentales mostraron un buen rendimiento en dos de los principales índices bursátiles de la India. Tras el análisis de los resultados

obtenidos los autores llegaron a la conclusión de que las estrategias combinadas, superaban a los mejores resultados obtenidos por cada una de las estrategias individuales, y también a los resultados de B&H en dos de los mayores índices de valores del mercado indio. El período de tiempo elegido, para el entrenamiento, iba del 1 de enero de 2003 al 31 de diciembre de 2009.

Machado et al. (2015) [132] utilizaron una metodología basada en el uso de la PG, y apoyada en indicadores técnicos, para desarrollar reglas de inversión siguiendo una perspectiva basada en el seguimiento de tendencias. Para mejorar la robustez de la solución del sistema resultante combinaron tres marcos de tiempo, con diferentes ponderaciones, para formar la posición final de mercado de cada día: largo plazo (LP), medio plazo (MP) y corto plazo (CT). Las pruebas se realizaron con una ventana móvil fijada en seis años y una ventana de muestra de datos bursátiles de seis meses, a partir del índice de bolsa estadounidense S&P 500. En sus resultados obtuvieron tasas de rendimiento anualizadas superiores al 10% en algunas configuraciones.

Por último, Pimenta et al. (2017) [165] propusieron un sistema que empleaba: el análisis técnico; un procedimiento de filtrado de valores atípicos; un método de selección de características; técnicas de programación genética y conjuntos de decisión, para obtener reglas de inversión eficientes. La primera fase comprendía el filtrado de los valores atípicos, puesto que, en palabras de los autores, “en las series de tiempo financieras, los valores atípicos son variaciones anormales del precio de la acción, que son a menudo una indicación de un factor externo que no se podría modelar sin información privilegiada”.<sup>7</sup> Para ello utilizaron un mecanismo que se apoyaba en el suavizado de puntos de dispersión ponderados localmente [39] (Locally Weighted Scatterplot Smoothing (LOWESS)). La segunda fase tenía como objetivo la selección apropiada de indicadores técnicos, con la finalidad de minimizar el espacio de soluciones del problema. La tercera, y última fase, ejecutaba el entrenamiento de las reglas de inversión mediante un algoritmo de PG. Las reglas obtenidas por la PG se utilizaban para construir un conjunto, cuyo comité de decisión estaba formado por los individuos que componían la aproximación final del conjunto de Pareto entregado por el algoritmo. En sus conclusiones, los autores mostraron que su sistema pudo obtener rendimientos financieros considerablemente por encima del precio de variación de las acciones. También pusieron

---

<sup>7</sup> Los valores atípicos son observaciones que no siguen el patrón de la mayoría de los datos. [42]

en valor que su solución no se basaba en las características específicas de ningún mercado de valores, y que se podía aplicar fácilmente a otros tipos de series de tiempo financieras, con pequeños cambios en el ajuste de algunos parámetros.

*Esta página se ha dejado en blanco intencionadamente.*



## Capítulo 3

# Computación evolutiva

La computación evolutiva es una rama de la computación, y de la inteligencia artificial, que comprende, entre otros, a los métodos de búsqueda y aprendizaje automatizado, inspirados en los mecanismos de la evolución y selección natural propuestos por Darwin [48], como medio para explicar la evolución biológica<sup>1</sup>. Se han propuesto diversas aproximaciones a la computación evolutiva, entre las que podemos citar: las estrategias evolutivas [175], los algoritmos genéticos [92], los clasificadores genéticos [28], la programación evolutiva [67], la programación genética [112] y las gramáticas evolutivas [181] entre otras. Estos algoritmos se han utilizado ampliamente para resolver problemas en distintas ramas de conocimiento, entre las cuales la de la economía que nos ocupa. En este capítulo vamos a tratar dos de ellas la PG y las GE.

### 3.1 Programación genética

La programación genética es una técnica de optimización Metaheurística<sup>2</sup>, o búsqueda estocástica, en la que una población de programas (individuos) se me-

---

<sup>1</sup> A diferencia del darwinismo, donde la evolución es sólo un resultado de la competencia y la selección, en el lamarckismo los propios organismos controlan la evolución. La posibilidad de que se adquieran conductas o marcas adquiridas de padres a hijos está en serio debate y es motivo de estudio de la epigenética “Los genes también tienen una especie de memoria” [94]. El proceso es inverso al de la evolución, en el sentido de que los cambios adquiridos en el fenotipo son representados en el genotipo [91,213], como paso previo para traspasarse a las siguientes generaciones. En la PG se han utilizado técnicas lamarckianas en sistemas híbridos para optimizar la búsqueda local, por medio de gradientes, trabajando sobre los valores de los nodos hoja numéricos [209].

<sup>2</sup> La metaheurística pertenece a un tipo de optimización estocástica, en la que se utiliza algún grado de aleatoriedad en la búsqueda de soluciones a problemas difíciles.

jora iterativamente, de acuerdo con una función objetivo, también conocida como función de aptitud o función (*fitness*). El proceso se centra en buscar que, en las sucesivas generaciones, los individuos de la población contengan programas que proporcionen cada vez mejores soluciones. Dado que las estructuras susceptibles de evolución son los mismos programas, se trata de una forma de exploración, o inducción de programas, relacionada principalmente con la programación automática.

Aunque el algoritmo tiene varias limitaciones, como por ejemplo un alto coste computacional, o el hecho de que no pueda garantizar la obtención de óptimos globales, tiene en cambio ventajas muy importantes. Entre ellas podemos mencionar: la capacidad de ser fácilmente paralelizable; el poder trabajar con grandes espacios de solución y complejas funciones de aptitud (*fitness landscapes*); que facilita escapar de mínimos locales; que es fácil adaptarlo a diferentes problemas y, sobre todo, que ofrece soluciones que son interpretables.

La mayoría de estos rasgos se comparten con otras alternativas de computación evolutiva como son: los algoritmos genéticos [92], las estrategias evolutivas [175, 185] o la evolución diferencial (ED) [202], por nombrar algunas. Sin embargo, la PG ofrece una ventaja clave frente a las demás técnicas, cuando se trata de estrategias de inversión, y es que la estructura de las reglas no tiene que establecerse de antemano. En lugar de optimizar los parámetros de reglas que han sido previamente definidas, la PG evoluciona la regla completamente.

El núcleo básico de la técnica, sobre la que se basa el algoritmo de la PG, comienza con la inicialización de un conjunto de individuos, o programas, a partir de un grupo de primitivas disponibles. A continuación, se evoluciona la población sucesivamente hasta que se cumple una determinada condición de parada. Por último, el resultado obtenido será el mejor individuo, o conjunto de individuos, encontrados hasta ese momento. Los pasos interiores del algoritmo son los que se enumeran a continuación:

- Ejecución del individuo (programa candidato) y evaluación de su aptitud mediante una función de aptitud determinada.
- Selección de un conjunto de individuos de la población siguiendo algún criterio basado en su aptitud. Este paso reduce la diversidad y dirige la población hacia aquellos individuos que son más aptos.

- Obtención de los nuevos individuos mediante la aplicación de algún operador genético (cruce, mutación, reproducción) a los individuos seleccionados. El objetivo de este paso es crear la diversidad necesaria, para facilitar la exploración del espacio de individuos factibles que se adecuen a la solución buscada.
- Reemplazo de todos o parte de los antiguos individuos de la población, en cada generación, con los nuevos individuos, siguiendo algún criterio basado en su aptitud.

### 3.1.1 Representación

La tarea de representación consiste en codificar al individuo (en el caso que nos ocupa una regla de inversión) en su forma cromosómica, es decir, en definir la aproximación que se sigue para construir la forma genotípica de la regla de inversión, donde se organiza la disposición de los genes en una estructura de datos que la soporta. La elección de la representación es de suma importancia, pues por un lado tiene consecuencias importantes en las funciones, u operadores genéticos, que modifican o tratan internamente al individuo [128] como son: la inicialización, la mutación, el cruce, la copia y la función de aptitud. Por otro lado, la búsqueda del óptimo sólo será posible, si el genotipo solución del individuo está incluido dentro de todas las representaciones posibles que este puede adoptar.

Que una técnica de CE tenga éxito recae en gran medida en la representación de los individuos y como estos son modificados, ya que cómo se modifique a estos, tiene un gran impacto en la forma que sigue la trayectoria de optimización en el paisaje de soluciones, y por tanto, en la calidad de la función de aptitud que los evalúa [219]. Si la representación es correcta se conseguirá que individuos similares tengan un comportamiento parecido, pues las modificaciones que sufran en su genotipo tendrán un efecto proporcional en su fenotipo. Cuando este es el caso, es posible que esos pequeños cambios en el fenotipo se vean reflejados por medio de pequeños cambios en la función de aptitud.

Una regla heurística a seguir, para conseguir calidad en la función de aptitud, consiste en aproximar el genotipo al fenotipo el máximo posible. Si el genotipo se codifica en forma de árbol, el fenotipo se ha de intentar que sea también representado de igual manera.

La PG representa tradicionalmente a los individuos como estructuras de árboles sintácticos. La representación de un individuo en forma de árbol tiene la ventaja de que se le puede evaluar fácilmente recorriendo el árbol, de forma recursiva en preorden, para conseguir una expresión prefija (notación polaca<sup>3</sup>) de árboles de expresión. Desde una expresión prefija se pueden aplicar, de forma simple, las operaciones indicadas en sus nodos no terminales sobre los operandos de los nodos finales, y sobre los resultados que se van generando en los nodos intermedios, por medio de un autómata de pila. Los árboles tienen, además, una correspondencia directa con las expresiones-s del lenguaje de programación lisp. Cada nodo no terminal del árbol tiene una función de operador y cada nodo terminal contiene un operando. Otra característica importante es que las expresiones matemáticas que contiene el árbol suelen ser simples y fáciles de evaluar. Por último, es sencillo aplicar al árbol del individuo los diferentes operadores genéticos tradicionales, mediante una pequeña adaptación. Por ejemplo, una copia de un individuo, o clonación, se traduce en la operación de recorrerlo en preorden insertando los valores en el nuevo árbol.

La figura 3.1 muestra un ejemplo de una representación en árbol de una regla de inversión, cuya expresión-s es:

$$(\text{Or } (>\text{Mx2 } \text{M2}) (\text{Or } (>\text{Min } \text{UR}) (<\text{M10 } \text{LR})))$$

La expresión anterior se puede interpretar fácilmente como: cuando el promedio móvil calculado de dos meses (M2) está por debajo del segundo indicador de resistencia del promedio móvil máximo de los últimos tres meses (Mx2), o cuando el mínimo de la sesión es mayor que la línea de tendencia superior (UR), o el promedio de diez meses (M10) es inferior a la línea de tendencia inferior (LR), entonces la mejor estrategia a seguir es entrar en el mercado y, en caso contrario, abandonarlo.

La figura 3.2 ilustra el comportamiento de la regla durante el año 2007. Allí podemos seguir la evolución del índice S&P 500. La línea más clara (verde) representa cuando la regla sugiere estar en el mercado o cuando recomienda permanecer en efectivo. Como podemos ver, la regla aconseja mantenerse fuera del

<sup>3</sup> En 1951, el lógico polaco Jan Lukasiewicz [125] sugirió que podía usarse una notación prefijada para todas las operaciones, de forma que se podría escribir  $+AB$ ,  $\subset AB$ , y señalaba que un uso consistente de tal notación eliminaba la necesidad de puntuación [19].

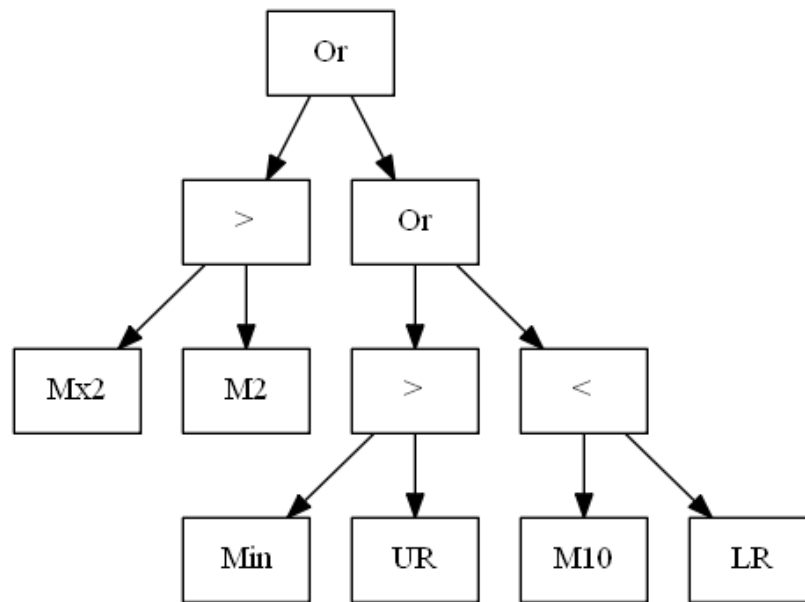


Figura 3.1: Regla de inversión de ejemplo, con su representación en árbol correspondiente.

mercado durante aproximadamente el primer mes y luego la recomendación varía con el tiempo. De acuerdo con la estrategia definida por la regla el inversor debe invertir durante el último mes. El hecho de que la regla sugiera un número limitado de transacciones contiene los costes de operación y, por lo tanto, aumenta las posibilidades de superar el índice.

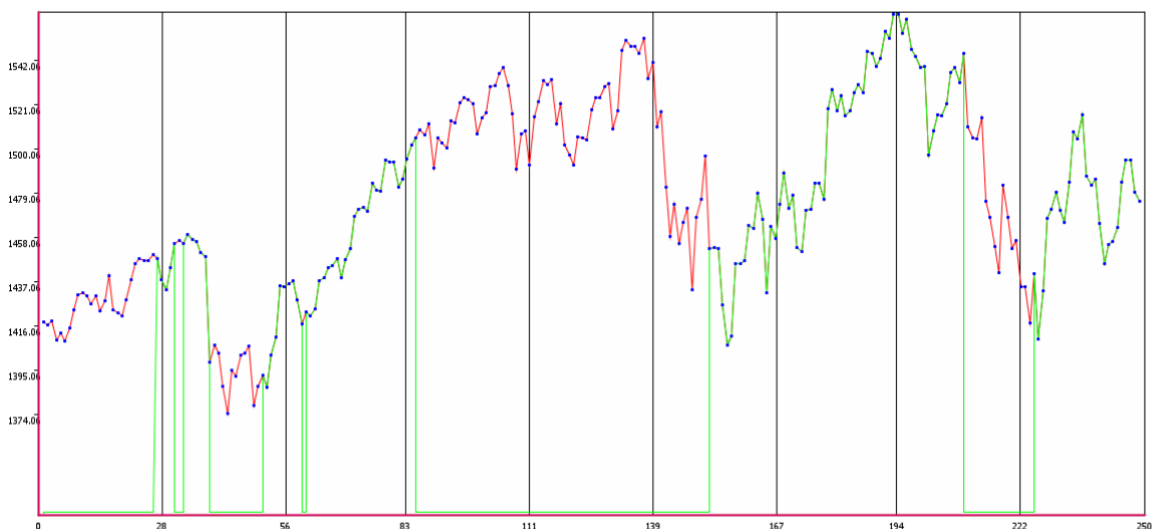


Figura 3.2: Comportamiento de la regla (Or (>Mx2 M2) (Or (>Mínimo UR) (<M10 LR))) en 2007.

### 3.1.2 Inicialización

La inicialización de la población en la PG sigue el mismo proceso aleatorio, de generación de los individuos de la población inicial, que usan otros algoritmos evolutivos.<sup>45</sup>

Aunque existen diferentes métodos para la inicialización de una población (*Ramped*, *Half-and-half*, *PCT1*, *PTC2*, *Randombranch* o *Uniform*), Sean y Liviu [130], en su estudio sobre el rendimiento de diferentes algoritmos de inicialización, concluyeron que estos proporcionan una variedad de ventajas para los investigadores, pero la evidencia experimental sugería que la mejora en la aptitud de los individuos generados no era una de ellas. No obstante, el poder especificar el tamaño preferido del árbol permite al usuario crear una distribución dirigida a favorecer la obtención de buenos resultados iniciales, de igual manera que el control en la operación de mutación del tamaño de los subárboles producidos, es una característica que permite evitar el crecimiento en complejidad de la solución. Algunos de los algoritmos, utilizados en un análisis preliminar de este trabajo, se exponen a continuación.

El primer algoritmo utilizado fue el de *Ramped half-and-half* (*RHH*). *RHH* es un algoritmo de inicialización utilizado en la PG, e ideado por Koza [112], para la creación de individuos en forma de árbol en la inicialización de la población. Esta basado en otros dos algoritmos conocidos como *Grow* y *Full*, también concebidos por Koza.

Tanto los algoritmos de PG *Full*, como *Grow*, crean a los individuos estableciendo límites en la profundidad máxima que los árboles que los representan pueden alcanzar. La profundidad máxima viene definida por el máximo número de aristas que se han de recorrer en el grafo del árbol, partiendo de su nodo raíz hasta llegar a cualquiera de sus hojas.

---

<sup>4</sup> La inicialización, no obstante, no siempre tiene que seguir un patrón aleatorio, sino que se puede inicializar un árbol de PG desde un punto determinista, llenando la población de inicio con un individuo que, aunque no sea una solución candidata, se cree que sea un buen punto de partida para alcanzar el individuo óptimo. Tal semilla puede haber sido producida por el resultado de un PG anterior o construida ad hoc por el investigador. Aler, Borrajo, Isasi [7]. Este enfoque es muy útil cuando se trata con poblaciones pequeñas donde hay poca variabilidad genética y es interesante dirigir la búsqueda por el espacio de soluciones.

<sup>5</sup> Algunos autores consideran que la elección de una inicialización dirigida debe tratarse con cuidado, eligiendo algunos individuos que no se correspondan con la heurística escogida. De esta manera se busca introducir cierta diversidad en el sistema, con el objetivo de evitar acabar en un mínimo local.

El método *Full* recibe su nombre por generar árboles de diferente tamaño, pero completos, es decir, que los árboles de los individuos creados tienen todas sus hojas a la misma profundidad, aunque tengan diferente número de nodos. En realidad esto sólo ocurre cuando en el conjunto primitivo de funciones utilizado, estas tienen todas la misma aridad<sup>6</sup>. La principal característica de esta inicialización es que el intervalo de tamaños de árboles, y por tanto formas de programa, producidos puede ser bastante reducido. Esto se cumple incluso cuando, como se ha comentado, en el conjunto primitivo de funciones algunas de ellas cuenten con diferentes aridades.

El método *Grow*, por el contrario, tiene un comportamiento más libre en la inicialización de la población, al permitir la creación de individuos cuyos árboles son de tamaños y formas más variadas. Los nodos se seleccionan a partir de todo el conjunto primitivo de funciones, tanto de terminales como de no terminales, hasta que se alcanza el límite de profundidad especificado. Una vez que se ha llegado a este límite, la elección del tipo de nodo seleccionado queda restringida a los elementos terminales.

El inconveniente de estos dos algoritmos es que, por separado, ninguno produce una variedad extensa de formas y tamaños de árbol. Para solucionarlo Koza desarrolló el algoritmo *RHH*, que se basa en la construcción de un árbol de una determinada profundidad. Su funcionamiento es sencillo, se escoge un valor aleatorio dentro del rango de profundidad establecido y, a partir de este valor, con una probabilidad determinada  $p$  se emplea el algoritmo *Grow* para generar el árbol, el resto de las veces según la probabilidad definida por  $(1 - p)$  el algoritmo emplea *Full*.

Aunque los tres algoritmos anteriores son fáciles de usar e implementar, la métrica de profundidad utilizada por *Grow*, *Full* y *RHH* es insuficiente para controlar la distribución estadística de los tamaños y formas de los árboles, que se pretende generar. Un ejemplo de este problema es la sensibilidad de los algoritmos al tamaño de los conjuntos de terminales y no terminales, utilizados en la creación del árbol. Si el tamaño del conjunto de terminales es mayor que el de los no terminales, siempre tenderá a crear árboles muy cortos, independientemente del límite de

---

<sup>6</sup> La aridad de un operador matemático, o de una función, es el número de argumentos necesarios para que dicho operador o función se pueda calcular. En PG un nodo con una aridad de cero es considerado un nodo hoja, y los nodos con una aridad mayor o igual que uno son considerados nodos no terminales.

profundidad establecido, y, por el contrario, si el número de funciones es significativamente mayor el algoritmo se comporta básicamente como *RHH*. Para evitar el problema anterior algunos algoritmos, como *Uniform*, nos permiten especificar el porcentaje de veces que probabilísticamente seleccionaremos un nodo intermedio o uno terminal. En este trabajo haremos uso de esta capacidad, implementándola en la inicialización de la población.

El algoritmo *RHH* no tiene definida una complejidad computacional en función del tamaño de la población [130].

Otro algoritmo de inicialización conocido como Probabilistic Tree-Creation 1 (*PCT1*), fue concebido por Sean Luke [127] como una modificación del algoritmo *Grow*. Sean buscaba garantizar que los árboles generados se construyesen a partir de un tamaño esperado y finito del árbol. *PCT1* parte de un tamaño de árbol esperado y de un máximo de profundidad establecido. El algoritmo comienza calculando la probabilidad  $p$  de elegir un elemento no terminal, sobre uno terminal, para mantener el tamaño del árbol esperado  $E$ , como se expresa en la fórmula 3-1:

$$p = \frac{1 - \frac{1}{E}}{\sum_{n \in N} \frac{1}{|N|} b_n} \quad (3-1)$$

Dónde  $N$  es el conjunto de todos los no terminales, y  $b_n$  es la aridad de un no terminal  $n$ . A partir de este computo, que se puede precalcular inicialmente, el algoritmo procede a crear el árbol.

La complejidad computacional de *PCT1* es lineal, o casi lineal en el peor de los casos, dependiendo del tamaño esperado del árbol  $E$ .

El algoritmo de inicialización, al que se le conoce como *Randombranch*, fue ideado por Chellapilla [38]. Este método de generación de árboles toma como parámetro un tamaño de árbol preferido, y, a partir de este valor, garantiza que este será o de ese tamaño, o "algo más pequeño". *Randombranch* procede dividiendo sucesivamente el tamaño del árbol a construir, equitativamente, entre los hijos de un padre que no sean nodos terminales. El inconveniente que presenta este algoritmo es que muchas estructuras de árboles no pueden producirse por la propia naturaleza del mismo. Esto tiene como consecuencia que el algoritmo *Randombranch* sea



un método de inicialización muy restrictivo.

La complejidad computacional de *Randombranch*, al igual que la de *PCT1*, es lineal o casi lineal en función del tamaño del árbol deseado.

En este trabajo se utiliza el algoritmo de inicialización uniforme (*Uniform*). Este método, introducido por Bohm y Geyer-Schulz [27]<sup>7</sup>, recibe su nombre del hecho de que el árbol generado es uniformemente exacto. El algoritmo solicita el tamaño del árbol que se desea generar, e intenta garantizar que este se crea de manera uniforme dentro del conjunto de todos los árboles posibles que tienen ese tamaño, dado un conjunto determinado de funciones. Durante el tiempo de generación del árbol, el algoritmo *Uniform* selecciona un nodo padre, a partir de una distribución derivada de unas tablas previamente calculadas. Si el nodo elegido es un nodo no terminal, se asigna un tamaño del árbol a cada uno de los nodos hijo, también calculado de las tablas de distribución, y se llama recursivamente a *Uniform* hasta alcanzar el tamaño de árbol deseado.

Dado que en esta tesis las funciones no terminales contienen únicamente dos entradas, estamos tratando con programas de tipo binario, donde la profundidad máxima y mínima de los árboles viene dada por las fórmulas 3-2 y 3-3, el máximo número de nodos de profundidad  $k$  por 3-4, y el máximo número de programas representables por un árbol binario de un determinado tamaño y profundidad por 3-5 [115] multiplicado por todas las formas de árbol posibles de ese tamaño y profundidad.

$$Max = \frac{l + 1}{2} \quad (3-2)$$

$$Min = \log_2(l + 1) \quad (3-3)$$

Donde  $l$  representa el número de nodos.

$$S_k = 2^{(k+1)} - 1 \quad (3-4)$$

<sup>7</sup> Existen otros métodos de inicialización uniformemente exactos como el de Iba [97], basado en el algoritmo biyectivo de Alonso [9].

$$P_{\text{programas}} = |T|^{\binom{l+1}{2}} |F|^{\binom{l-1}{2}} \quad (3-5)$$

Donde,  $T$  representa el número de diferentes elementos terminales y  $F$  el número de diferentes funciones no terminales a partir de las cuales se construye el programa.

El algoritmo *Uniform* presenta una tendencia a crear árboles de tipo asimétrico con algunas hojas muy cercanas a la raíz, a diferencia de *RHH* cuyas hojas están en promedio a cierta distancia de la misma. Una inicialización de tipo uniforme puede ayudar a resolver problemas en los que las soluciones deseadas son asimétricas, y en donde algunas hojas del árbol son mucho más importantes que otras [166].

Durante la inicialización de la población la complejidad computacional de *Uniform* es casi lineal, en relación al tamaño del árbol elegido. No obstante, dado que antes de la construcción del individuo se han de calcular las tablas que utiliza posteriormente, como por ejemplo la tabla que contiene los números de árboles posibles para todos los tamaños de árbol admisibles, hasta llegar al tamaño máximo permitido, es un algoritmo más costoso que *Full*, *Grow* o *RHH*. Debido a ello, la complejidad de *Uniform* es superlineal, pero limitada a un factor de complejidad polinomial.

### 3.1.3 Operadores

La función de los operadores genéticos es ayudar en el proceso de exploración y explotación de los AG, en la búsqueda de soluciones, transformando la población inicial sucesivamente por medio de un proceso recursivo, hasta llegar a un estadio evolutivo que contenga un individuo que represente un resultado óptimo o satisfactorio.

En los AG, el proceso de generación de soluciones candidatas, en las nuevas poblaciones, es impulsado por la aplicación de los operadores genéticos de mutación y cruce. Estos se complementan con el operador de reproducción, que selecciona un elemento de la población existente y lo copia en uno nuevo sin modificar [77].

Aunque Koza identificó a estos tres operadores como los operadores básicos de la PG, en la mayoría de las aplicaciones sólo se usa el operador de cruce y

el de selección, por tanto en los pasos evolutivos del algoritmo no se introduce nuevo material genético. Sólo la recombinación de la carga genética producida en la inicialización es tenida en cuenta. El mismo Koza delegó la operación de mutación a un segundo plano, probablemente por ser un mecanismo de exploración a largo plazo [112]<sup>8</sup>. En este trabajo, si se hará uso de la mutación, empleándola en una pequeña cantidad.

En la implementación, que se utiliza en los experimentos de los diferentes planteamientos abordados en esta tesis, la aplicación de los operadores genéticos se gestiona por medio de tuberías de reproducción [112]. Estas tuberías conforman unas cadenas de operadores de selección y reproducción, cuya función es obtener individuos de una población existente para producir individuos en una nueva. Dado un determinado conjunto de probabilidades, que se asignan a las operaciones de cruce, mutación y reproducción, el proceso elige probabilísticamente a un operador y proporciona soluciones candidatas para la siguiente generación: mediante el cruce de varios individuos; la introducción de un individuo en la nueva población después de una mutación; o mediante la simple copia del individuo sin mutación. La tubería ha de cumplir dos condiciones: la primera es que todos los operadores genéticos involucrados tengan una probabilidad mayor de cero de ser seleccionados, y la segunda es que se determine, de forma precisa, dicha probabilidad para cada uno de los operadores. El uso de las tuberías en la PG es explotativo, pues permite que padres con una alta aptitud permanezcan en la población y compitan con sus hijos.

El operador de cruce genera soluciones, o descendientes, a partir de los padres. En la PG, al ser la representación del individuo, o genotipo, una estructura en árbol,

---

<sup>8</sup> Esto probablemente se debió a sus trabajos en la teoría de esquemas. En ella argumentaba que los individuos que contienen buenos esquemas tienen, en promedio, valores de aptitud más altos que los individuos que compiten dentro de la misma generación. Mayor aptitud resulta en una mayor probabilidad de reproducción. Siguiendo esta línea de razonamiento, los buenos esquemas vivirán y se recombinarán, por medio del operador de cruce, para lograr incluso mejores esquemas. Es más probable que el operador de cruce interrumpa los esquemas complejos, esto lleva a la evolución de esquemas pequeños, pero más potentes, conforme pasan las generaciones en PG, que se combinan por cruce sucesivamente para lograr soluciones aún mejores. Con el tiempo, esto guía a la PG a buscar partes más prometedoras en el espacio de soluciones. En palabras del propio Koza "... concentra la búsqueda del espacio de la solución en subespacios de las expresiones  $S$  de LISP de dimensionalidad cada vez menor y mejor aptitud". La solución final se desarrolla luego como una combinación de varios esquemas pequeños pero poderosos, que se conoce como "la hipótesis de bloques de construcción". Curiosamente, Koza no hace más comentarios sobre el operador de mutación y considera que el cruce es el único operador que impulsa a la PG. De hecho, usa  $p = 0$  en sus aplicaciones.

el cruce se puede realizar intercambiando ramas completas de los árboles, trozos de las mismas (injertos) o nodos aislados de cada uno de ellos. En este trabajo se aplicará el cruce de subárbol estándar en el que: se toman dos individuos o padres de la población, a continuación, se selecciona el árbol de cada uno de ellos, seguidamente se selecciona aleatoriamente un nodo en cada árbol, y finalmente se intercambian los dos subárboles arraigados por esos nodos. Se aplica, por tanto, el intercambio de ramas, pero no el de nodos o injertos.

Una consecuencia del mecanismo de cruce expuesto es que en la PG, a diferencia de los que sucede en los AG, dos padres cuya solución es idéntica pueden generar soluciones completamente nuevas. Para evitar que la variabilidad genética se reduzca se suele imponer que en cada obtención de una nueva solución candidata, y dentro del grupo de reemplazo, sólo se acepten aquellos individuos cuya carga genética no esté ya presente en el conjunto de intercambio. Puesto que esta solución puede producir situaciones en las que operador de cruce no pueda completar la tarea por excesivo número de colisiones, dejando al bucle del algoritmo de la PG bloqueado, se suele utilizar una técnica de mitigación para evitarlo, que consiste en reintentar el cruce con diferentes padres un número máximo de veces, aumentando dinámicamente la probabilidad de mutación de los descendientes si fuese necesario. Típicamente la mutación sigue una aproximación gaussiana auto-adaptativa [90], de forma que cuando la población se concentra en una región muy pequeña, lo que se determina por la varianza de la distribución, las mutaciones son reducidas y cuando los individuos de esta se dispersan esta es mayor. Cuando la convergencia es muy alta, es interesante cambiar este modelo por otro completamente opuesto, en el que la mutación aumenta para mantener las capacidades de exploración del algoritmo.

El operador de mutación, en la PG, puede actuar de dos formas: mediante la mutación de tipo *intranodo* o mediante la modificación de la estructura del árbol del individuo, *extranodo*.

La mutación *intranodo* actúa sobre el valor interno del nodo seleccionado, que será sobre una función en los nodos intermedios o sobre un terminal en los nodos hoja. Esta se puede implementar mediante un operador de mutación de tipo simple, en la que se selecciona un nodo al azar y se cambia por otro del mismo tipo (terminal o no terminal), generado aleatoriamente. Si se trata de un nodo terminal se muta por otro nodo terminal, y si este es un nodo intermedio se cambia por

otra función del mismo tipo (mismo tipo de retorno, igual número de argumentos y tipo de los mismos).

La mutación de tipo *extranodo*, se puede aplicar de diferentes formas: por *crecimiento*, por *decrecimiento* o por *intercambio*. En la mutación de *crecimiento*, el operador de mutación selecciona arbitrariamente una hoja del árbol y la sustituye por un subárbol aleatoriamente generado; en la de *decrecimiento*, se selecciona un nodo intermedio cualquiera del árbol y se sustituye por un subárbol, también aleatoriamente generado; y en la mutación de *intercambio*, se selecciona un nodo intermedio del árbol y se reordenan sus argumentos.

En esta tesis, se hace uso de la mutación *extranodo* de *crecimiento* y *decrecimiento* de la estructura del árbol, en la que se selecciona un nodo terminal o no terminal, del individuo seleccionado, y el subárbol que cuelga de él es reemplazado totalmente por un nuevo árbol generado aleatoriamente.

Los operadores de cruce y mutación son una parte muy importante del poder de la PG, que está basado en los llamados esquemas. La teoría de esquemas describe cómo varias combinaciones de genes evolucionan a lo largo de la ejecución de un algoritmo de PG. El teorema del esquema establece que el número de cromosomas que coinciden con un esquema aumentará, debido al efecto de la selección, si el esquema tiene una aptitud superior a la media, pero disminuirá como resultado de los efectos perturbadores de los operadores genéticos de cruce y mutación. No obstante, tanto el cruce como la mutación no sólo pueden interrumpir el esquema, sino que también pueden formar nuevos cromosomas que coincidan con el esquema. Si se considera el espacio de soluciones como un hypercubo [128], en donde cada uno de los individuos representa un punto del mismo, cuando se realizan cruces los nuevos individuos de la población se van desplazando hacia nuevas posiciones. Si en ese proceso, se pierden los alelos de algunos de los genes del cromosoma de un individuo, el resultado es una reducción del espacio de soluciones que representa el hypercubo. Cuando la población converge, la consecuencia probable es que todos los individuos se vayan dirigiendo hacia una posición específica del hypercubo, y en el extremo la población se reduzca a un punto, que si hay suerte es el correspondiente al óptimo. En el caso de una convergencia prematura este punto será probablemente un óptimo local.<sup>9</sup>

---

<sup>9</sup> Una descripción rigurosa de la teoría de esquemas se puede encontrar en Langdon y Poli (2002) [115].

La selección es el proceso por el que se elige a parte o todos los individuos de una población, con el objeto de crear un conjunto de nuevos miembros que se introducirá, parcial o totalmente, en una nueva población. El reemplazo, aplicado a los miembros de la población original, puede seguir diferentes estrategias: esta puede ser tal que el reemplazo sea total, parcial o sin reemplazo. El propósito de la selección es impulsar a los miembros más adaptados a la nueva población, con la esperanza de que estos tengan una mejor aptitud aún que sus progenitores y proporcionen, por tanto, una mejor solución. Para ello se ha de determinar cómo elegir los genes de los padres que formarán la descendencia, y cuál es el mecanismo de lucha que determina que progenitores elegir.

En las EE se utiliza un procedimiento de selección conocido como selección por *truncamiento*, en él son directamente elegidos los  $\mu$  individuos más aptos de la población, que van a formar de la siguiente generación, descartando el resto [128]. Los AG, debido a las características iterativas de la selección que se utiliza en su algoritmo, proporcionan otras alternativas. Típicamente se pueden distinguir dos esquemas de selección: la selección basada en un método proporcional (*ruleta*) y la selección basada en un método ordinal (*ranking, torneo*).

La selección proporcional elige a los individuos en función de su valor de aptitud, relativo a los otros miembros de la población, mientras que la selección ordinal lo hace, no en valor absoluto de la aptitud, sino en función del *ranking* del individuo dentro de la población. Esto requiere que la presión de selección sea independiente de la distribución del valor de aptitud dentro de la población, y se base únicamente en su orden relativo dentro de esta. [197]

Por ejemplo, en la selección proporcional por el mecanismo de *ruleta*, propuesta por Holland [92] y desarrollada por De Jong [24], a cada uno de los individuos de la población se le asigna una probabilidad de selección en la *ruleta* en función de su valor de aptitud, de tal forma que la suma de todas las probabilidades asociadas a cada uno de los individuos sea la unidad. Es un método que utiliza la probabilidad aplicada a la adecuación de la solución.

La selección por *ruleta* es el método de selección más fácil, simple de implementar y que consume la menor cantidad de tiempo computacional. Sin embargo, sufre de problemas de convergencia prematura, lo que lleva a menudo a encontrar soluciones óptimas a nivel local (mínimos o máximos locales).

En la selección por *ruleta* cuando los individuos son ordenados, para ser seleccionados, lo son de acuerdo a una medida de probabilidad basada en su función de aptitud, siguiendo la fórmula 3-6 [191], donde  $f(x_i)$  es el valor de aptitud del individuo  $i$  y  $n$  es el tamaño de la población.

$$Pr_i = \frac{f(x_i)}{\sum_{i=1}^{i=n} f(x_i)} \tag{3-6}$$

El tamaño asignado en la ruleta al elemento  $x_j$  se establece a partir de la probabilidad de selección acumulada según la ecuación 3-7.

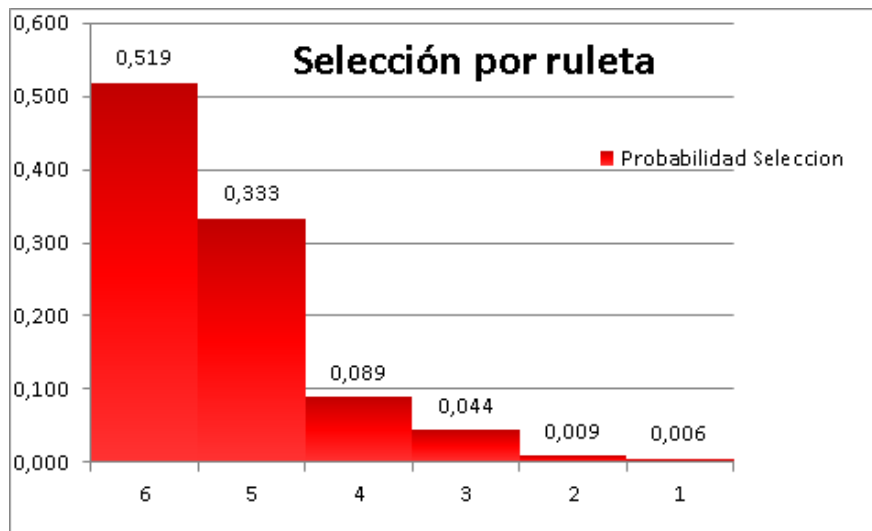
$$Pac_{x_j} = \sum_{i=1}^{i=j} Pr_i \tag{3-7}$$

INDIVIDUO	$f(x)_i$	$Pr_i = \frac{f(x)_i}{\sum_{i=1}^{i=n} f(x)_i}$
1	700,00	0,52
2	400,00	0,33
3	120,00	0,09
4	60,00	0,04
5	12,00	0,01
6	8,00	0,01
$\sum_{i=1}^{i=n} f(x)_i$	1350,00	Se presenta con redondeo.

Figura 3.3: Tabla de selección de individuos por el sistema de ruleta.

Gráficamente, para seis individuos, se podría representar el método de selección por *ruleta* de acuerdo a los valores de la tabla del figura 3.3 mediante el gráfico de la figura 3.4.

El mecanismo de selección por *ruleta* no es adecuado cuando existen considerables diferencias entre los valores de ajuste de los diferentes individuos de la población. Por ejemplo, si un único individuo ocupa un sector cuyo tamaño es 70% de la *ruleta*, el resto de los individuos tendrán muy pocas posibilidades de ser elegidos. En este caso una selección de tipo *ranking* sería mucho más apropiada.



**Figura 3.4:** Modelo de probabilidad de selección de individuos por el sistema de ruleta.

En la selección por *ranking*, introducida por Baker [14]<sup>10</sup> para eliminar los problemas de la selección proporcional, la elección del individuo depende de su *ranking* absoluto dentro de la población. El efecto obtenido es que se produce una repartición más uniforme de la probabilidad de selección entre los diversos individuos participantes. Para una población de tamaño  $n$  al individuo con menor *ranking* se le asignará un *ranking* cuyo valor es  $r = 1$ , y el que tenga la mejor aptitud se le asignará el valor  $r = n$ . El método mantiene una presión de selección consistente y la estrategia de selección es robusta. Sin embargo, la convergencia es lenta, al ser la aptitud de los individuos que participan en la reproducción muy similar entre sí [198].

La probabilidad de selección se asigna linealmente a los individuos de acuerdo a su rango, y viene dada por la fórmula 3-8 [26,37,193].

$$P_{s_i} = \frac{1}{n} \left( \min + \frac{(\max - \min)(\text{rank}(i) - 1)}{(n - 1)} \right) \quad (3-8)$$

Donde, si  $\max + \min = 2$  y  $1 \leq \max \leq 2$ , entonces se cumple que la probabilidad de la distribución es  $\sum_{i=1}^{i=n} P_{s_i} = 1$ . La probabilidad del peor individuo vendrá

<sup>10</sup> Existe una variante de la selección proporcional por el método de ruleta, propuesta por el propio Baker [15], conocida como muestreo estocástico universal (Stochastic Universal Sampling (SUS)), en la que los individuos son seleccionados al menos una vez, y que además reduce la complejidad de un orden polinómico  $O(n^2)$  a uno logarítmico  $O(\log(n))$ .



dada por  $P_{min} = \frac{min}{n}$ .

En el caso de que los parámetros de selección se establezcan en  $max = 2$  y  $min = 0$ , la probabilidad  $P_{s_i}$  entonces viene dada por la ecuación 3-9, que establece la presión más alta (el peor individuo no se seleccionará nunca).

$$P_{s_i} = \frac{2 \cdot rank(i) - 2}{n \cdot (n - 1)} \quad (3-9)$$

Si  $max = 1$  y  $min = 1$ , la probabilidad  $P_{s_i}$  de selección se define de acuerdo a 3-10, y esta será completamente aleatoria.

$$P_{s_i} = \frac{1}{n} \quad (3-10)$$

Y por último, si  $max = 1,5$  y  $min = 0,5$ , la probabilidad de selección  $P_{s_i}$  viene dada por 3-11, que como se puede apreciar en el gráfico de la figura 3.5, proporciona una selección lineal, y tiene la característica de que siempre otorga la posibilidad de que cualquier individuo sea seleccionado, incluso los individuos más débiles de la población.

$$P_{s_i} = \frac{0,5}{n} + \frac{rank(i) - 1}{n \cdot (n - 1)} \quad (3-11)$$

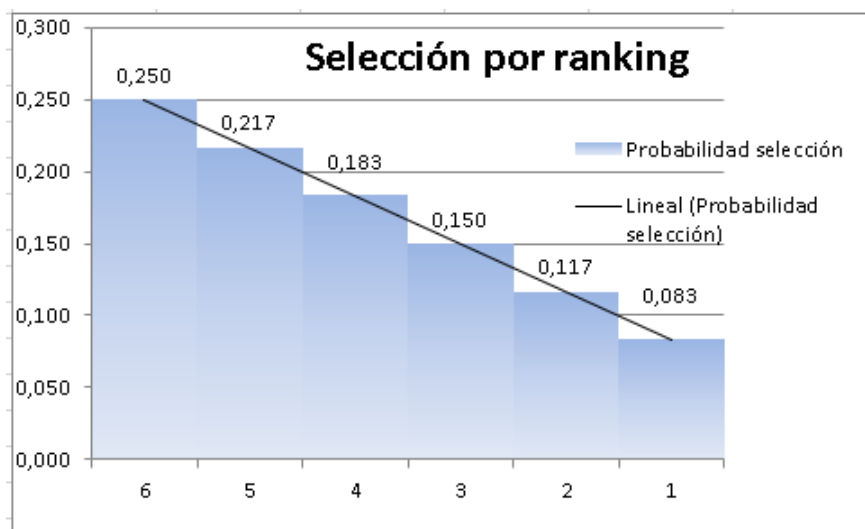


Figura 3.5: Modelo de probabilidad de selección de individuos por el sistema de ranking.

La selección por *ranking* presenta el problema de que cuando el algoritmo está convergiendo, y los individuos están muy próximos unos a otros, en su valor de aptitud, tendrán todos prácticamente la misma posibilidad de ser seleccionados, y el algoritmo convergerá al final efectuando básicamente una selección aleatoria. Además de una convergencia prematura, el método de *ranking* tiene otro inconveniente, y es que este tipo de selección presenta un rendimiento computacional, que, aunque mayor que el de ruleta se puede mejorar. Un método que supera en rendimiento computacional al de *ranking* es el de *torneo*, ya que su velocidad de proceso es mayor<sup>11</sup>, al no ser necesario ni evaluar ni comparar la totalidad de la población, y presenta además la capacidad de prevenir, en un cierto grado, una convergencia prematura.

En la estrategia de selección por *torneo* [138] lo que se trata es de seleccionar un individuo, entre varios, mediante la comparación directa de los valores de aptitud de todos ellos. Existen dos estrategias: una determinista en la que se elige un número de individuos al azar, normalmente dos, y de entre ellos el más apto pasa al grupo de padres que establecerán el cruce; y otra probabilística que se diferencia de la determinista en el paso de selección del ganador del torneo. En el método probabilístico, en vez de escoger siempre aquel individuo con mejor aptitud, se establece una probabilidad de ser ganador basada en ese mismo valor de aptitud. Es posible ajustar la presión de selección cambiando el tamaño del *torneo*, y dado que la selección del *torneo* da oportunidades a todos los individuos de competir, preserva en cierto modo la diversidad.

La probabilidad de ser elegido para el *torneo* (suponiendo un torneo entre dos individuos, donde además puede haber repetición al existir reemplazo) viene dada por la fórmula 3-12.

$$P_s = \frac{n}{\binom{n+2-1}{2}} = \frac{2}{n+1} \quad (3-12)$$

Donde  $n$  es el tamaño de la población.

Por otro lado la probabilidad de que gane el *torneo*, a otro miembro selecciona-

<sup>11</sup> La complejidad computacional se reduce de  $O(n \log n)$  del método de *ranking* en el mejor de los casos, y  $O(n^2)$  en el peor, a  $O(n)$ . [78]

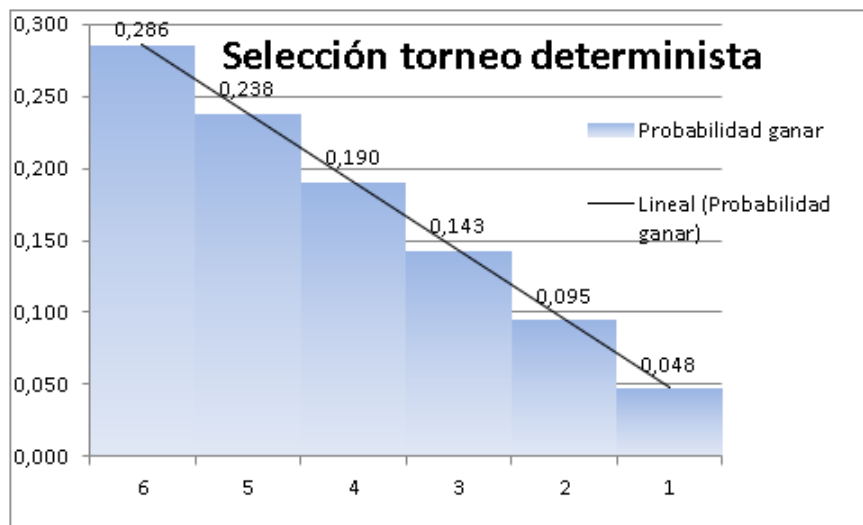
do de menor o igual rango, cuando hay posibilidad de repetición de un individuo al existir reemplazo, es igual al valor obtenido de la fórmula 3-13.

$$Pg_i = \frac{rank(i)}{n} \tag{3-13}$$

Con lo que la probabilidad de ser seleccionado para ser integrado en la nueva población es el resultado de la ecuación 3-14.

$$Ps_i = \frac{2 \cdot rank(i)}{n \cdot (n + 1)} \tag{3-14}$$

En el gráfico de la figura 3.6 se puede observar el comportamiento estadístico de la selección por *torneo determinista*.



**Figura 3.6:** Modelo de probabilidad de selección de individuos por el sistema de torneo determinista de dos individuos.

El número de participantes en el torneo establece la presión de selección, que aumenta de forma proporcional a su tamaño. En el límite inferior, en el que  $T = 1$ , el torneo se comporta como una búsqueda aleatoria favoreciendo la exploración. Conforme  $T$  aumenta el operador genético de selección dirige el algoritmo hacia la explotación. Es posible establecer un mecanismo probabilístico en la elección de esta presión, de forma que el valor de  $T$  dependa de él. El valor utilizado más popular en la literatura es de dos individuos [32]. Para determinarlo es posible

acudir a la fórmula 3-15 que fija el número de descendientes  $D$ , que producen los individuos cuyo rango es inferior a  $R$ , para un torneo de tamaño  $T$ . [25,115]

$$D = n \cdot \sum_{i=1}^R \frac{\left(\frac{\text{rank}(i)}{n}\right)^T - \left(\frac{\text{rank}(i-1)}{n}\right)^T}{\text{rank}(i) - \text{rank}(i-1)} \approx n \left(\frac{R}{n}\right)^T \quad (3-15)$$

En el peor de los casos, la mitad de la población de una generación es producida por  $\sqrt{1/2}^n$  individuos de la generación anterior [115]. Si eligiéramos un valor de  $T$  de 8 individuos, el 8.3 % de la población de una generación sería responsable del 50 % de los mejores individuos de la siguiente generación. Esto sería excesivo pues muchos de los individuos serían idénticos a sus padres, estableciéndose una estrategia elitista enorme. Si, por otra parte, el valor de  $T$  fuese 2 la presión se reduciría considerablemente, al ser la cantidad responsable de la mitad de los individuos de la siguiente generación el 29.3 %, la tercera parte aproximadamente.

Cuando el número individuos que participan en el torneo es alto la presión de selección es elevada, y los peores individuos tendrán pocas opciones de ser seleccionados para reproducirse. En el extremo, cuando todos los individuos participan en el torneo, estaríamos ante lo que se conoce como elitismo global. En este caso la presión del medio es máxima y la selección se convierte en completamente determinista. Por el contrario, cuando el tamaño de los participantes en el torneo es reducido la presión de selección disminuye y los peores individuos tienen más oportunidades de ser seleccionados. En el límite, cuando el número de participantes en el torneo es de 2, la presión del medio es mínima, si a ello añadimos una estrategia probabilística con  $Ps = 0,5$  estaríamos ante un sistema que ofrece la misma presión a cualquier individuo. Si  $Ps$  fuera igual a 1 no estaríamos ante una estrategia de torneo sino ante una selección determinista.

La elección de uno u otro método de selección determinará la estrategia de búsqueda del algoritmo genético. Un método con una alta presión de selección centra la búsqueda de las soluciones en un entorno próximo a las mejores soluciones con las que cuenta la población en curso, corriendo el riesgo de que esta se encuentre alrededor de un mínimo local. Por el contrario, optando por una presión de selección menor se deja el camino abierto para la exploración de nuevas regiones del espacio de soluciones.

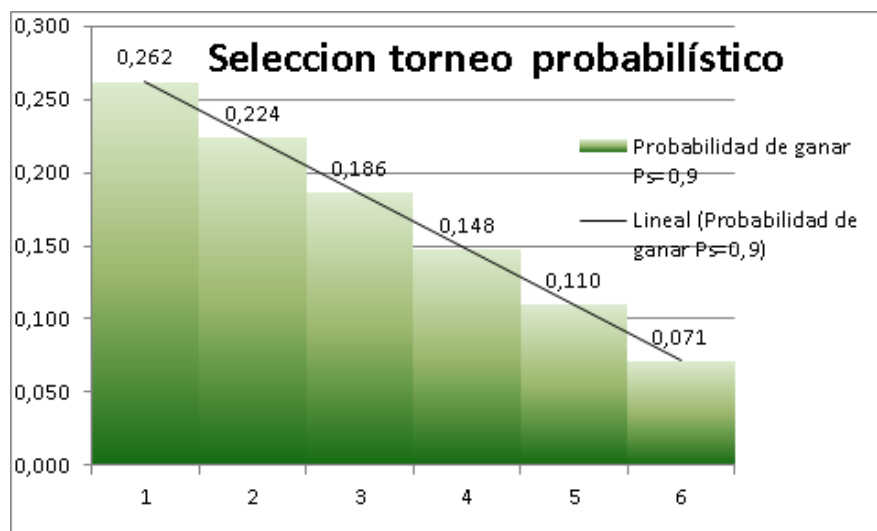
En el caso probabilístico, la probabilidad de selección de los individuos que

forman parte del torneo es la misma, pero la probabilidad de ganar a otro contrincante y la probabilidad total de ser seleccionado para la nueva población viene dada por las fórmulas 3-16 y 3-17.

$$Pg_i = \frac{rank(i)}{n} \cdot P_i + \frac{rank(i)}{n} \cdot (1 - P_i) \quad (3-16)$$

$$Ps_i = \frac{2 \cdot rank(i)}{n \cdot (n - 1)} \cdot P_i + \frac{2 \cdot (n - rank(i) + 1)}{n \cdot (n - 1)} \cdot (1 - P_i) \quad (3-17)$$

Gráficamente para 6 individuos se podría representar la estrategia probabilística del método de torneo, cuando  $Ps = 0,9$ , como se ve en la figura 3.7.



**Figura 3.7:** Modelo de probabilidad de selección de individuos por el sistema de torneo probabilístico de dos individuos.

En este trabajo se utilizará la estrategia de selección por *torneo* determinista, en la que se eligen dos individuos completamente al azar con reposición, para formar el conjunto de participantes en el mismo. En esta variante de la estrategia de selección por torneo puede ocurrir que el mismo individuo sea elegido más de una vez para competir, al ser introducido de nuevo en la población tras la fase de selección previa. Para que la estrategia de selección por *torneo* funcione correctamente, se ha de cumplir la condición de que la función de aptitud utilizada adopte valores no negativos. El resultado final del proceso devolverá el individuo más apto de los que entren en liza, para ser proporcionado al mecanismo de reproducción.

Por último, la operación de reproducción se encarga de la sustitución de parte o todos los individuos de la antigua generación, por otros nuevos creados mediante el operador de cruce. Existen básicamente dos estrategias de reproducción: la basada en algoritmos generacionales, y la de tipo estacionario (steady-state) [212].<sup>12</sup>

En el reemplazo generacional, toda la población de genomas se reemplaza en cada generación por medio de la creación de una población intermedia. Eso significa que los recién nacidos engendrados avanzarían a la siguiente generación y reemplazarían a todos los individuos de la generación en curso.

En los sistemas generacionales hay diversas estrategias que se pueden aplicar. Por ejemplo en la  $(\mu, \lambda)$  [175] se comienza con una población de  $\mu$  individuos. A continuación, la población de  $\mu$  padres se utiliza para crear  $\lambda$  hijos. Para completar el número  $\lambda$  de hijos cada uno de los individuos padres de  $\mu$  tiene que producir  $\mu/\lambda$  nuevos individuos. El valor  $\mu$  representa a los padres de la generación previa, que se usan para procrear, y  $\lambda$  a los hijos que estos producen. Los descendientes creados son clasificados de acuerdo con su valor de aptitud, y los mejores  $\mu$  se seleccionan para reemplazar la población de cromosomas parentales [186]. En este caso, sólo la descendencia se está utilizando para reemplazar a la población parental. El proceso continua iterativamente hasta alcanzar la condición de parada establecida.

En un segundo algoritmo de ejemplo, que sigue la estrategia generacional, conocido como  $(\mu + \lambda)$ , cada una de las siguientes generación en el ciclo evolutivo se compone de los  $\mu$  padres, a los que se unen los nuevos hijos generados  $\lambda$ . En cada generación los padres producen  $\lambda$  hijos, que compiten con ellos para ocupar el espacio de los  $\mu$  individuos eliminados.

En términos generales el algoritmo  $(\mu + \lambda)$ , en comparación con  $(\mu, \lambda)$ , explota más que explora el espacio de soluciones, porque los padres con más aptitud sobreviven para competir con sus hijos indefinidamente. Esto tiene riesgos, pues un padre suficientemente apto puede derrotar a otros miembros de la población una y otra vez, causando finalmente que toda la población converja en la dirección

---

<sup>12</sup> Ambas estrategias se encuentran en la naturaleza y se comportan de la misma forma. Por ejemplo, muchas plantas siguen un mecanismo generacional en el que sólo las semillas sobreviven al invierno. En la estrategia estacionaria seguida por multitud de especies, por otra parte, los individuos se van creando continuamente siendo los sustitutos, a medio plazo, de otros individuos dentro de la población, que crece o decrece conforme haya nuevos nacimientos o defunciones.

de los descendientes inmediatos de ese progenitor. En esta situación, las soluciones representadas en la población pueden acabar en un óptimo local que rodee al padre.<sup>13</sup>

En el reemplazo seguido por la estrategia estacionaria no se crea una nueva población en cada ciclo evolutivo, sino que  $n$  individuos de cada iteración son seleccionados en grupos (normalmente de dos en dos), que se cruzan entre ellos para crear  $n$  individuos de descendencia y estos reemplazaran a  $n$  individuos de la población original. Uno por uno, los individuos se generan y luego son enviados a la función de aptitud para ser evaluados. En la evolución estacionaria esta evaluación se realiza al instante y se devuelve a cada individuo inmediatamente. Una vez que el individuo ha sido evaluado otro miembro de la población existente se marca para ser reemplazado. Por último, se sigue una de las dos acciones siguientes: con una cierta probabilidad el individuo marcado es reemplazado directamente por el nuevo individuo, y en caso contrario este será reemplazado por el nuevo individuo, pero sólo si la aptitud de este último es mayor.

### 3.1.4 Conjuntos de elementos terminales y no terminales

En la PG el árbol del individuo, genotipo o cromosoma<sup>14</sup>, que representa la solución, contiene un programa de ordenador definido sobre un lenguaje de programación. Cada uno de los nodos de los que se compone el árbol incluye a un elemento de programación que puede ser de uno de los dos tipos siguientes: uno que conforma a los terminales, que ocupan los nodos finales u hojas del árbol, y otro que constituye a los no terminales, que se localizan en los nodos intermedios o nudos de este.

Los nodos no terminales pueden pertenecer a alguno de los siguientes elementos:

- Entradas externas del programa. Estas toman típicamente la forma de variables:  $x$ ,  $y$ ,  $z$ ...

---

<sup>13</sup> Esto es fácilmente solucionable estableciendo un límite, o factor de envejecimiento, que penalice al padre en función de su edad evolutiva. Esta misma estrategia es la seguida por la propia naturaleza mediante un mecanismo de obsolescencia programada que, añadida a la entropía de los procesos biológicos, provoca el envejecimiento y la pérdida de aptitudes de los individuos.

<sup>14</sup> Más bien, por su complejidad, se puede considerar al árbol del individuo, en programación genética, al nivel biológico de una proteína.

- Funciones sin argumentos.
- Constantes del programa. Predefinidas, o generadas al azar en la inicialización, como es el caso de las ERC.

En cuanto a los nodos no terminales, que constituyen el conjunto de funciones del lenguaje usado, pueden ser entre otros de los siguientes tipos:

- Operadores binarios.
- Operadores aritméticos.
- Operadores lógicos.
- Operadores relacionales.
- Funciones matemáticas o funciones más específicas relacionadas con una aplicación de la PG en particular.

### 3.1.5 Función de aptitud

La manera en la que la PG guía a la población de soluciones candidatas, en el proceso de búsqueda de una solución que sea óptima, mediante la exploración y explotación del espacio de soluciones, es a través del uso de la función de aptitud. La función de aptitud ayuda a discernir que regiones del espacio de búsqueda proporcionan individuos (programas, reglas, ...) idóneos, que cumplen con los requisitos de la función objetivo, o en otras palabras; la medida de aptitud es el mecanismo principal para dar una declaración de alto nivel, de los requisitos del problema, al sistema de PG.

La aptitud se puede medir de formas diferentes. Cuando se conoce la salida esperada, se puede expresar mediante una medida de error por medio de algún método como: el de mínimos cuadrados, el de máximo error o el de suma del valor absoluto del error.

- Mínimos cuadrados:

$$J(f_1, f_2, \dots, f_k, P_1, P_2, \dots, p_l) = - \sum_{i=1}^{i=n} (y_i - g(x_i, f_1, f_2, \dots, f_k, P_1, P_2, \dots, p_l))^2 \quad (3-18)$$



- Suma del valor absoluto del error:

$$J(f_1, f_2, \dots, f_k, P_1, P_2, \dots, p_l) = - \sum_{i=1}^{i=n} |(y_i - g(x_i, f_1, f_2, \dots, f_k, P_1, P_2, \dots, p_l))| \quad (3-19)$$

- Máximo error:

$$J(f_1, f_2, \dots, f_k, P_1, P_2, \dots, p_l) = - \max_{1 \leq i \leq n} |(y_i - g(x_i, f_1, f_2, \dots, f_k, P_1, P_2, \dots, p_l))| \quad (3-20)$$

Sin embargo, esta salida no siempre es conocida y cuando esto ocurra habrá que tratar con el valor obtenido de la función de aptitud directamente. Si como en el caso que se pretende tratar, lo que se busca es generar un programa que permita predecir, a partir de una serie temporal de datos y mediante una regla de inversión, el comportamiento futuro del mercado, en ese caso, la conformidad de una estructura de datos de entrenamiento con la que presenten los datos futuros sólo se podrá resolver con la PG, si los criterios de adecuación de la función de aptitud son elegidos correctamente. La elección de esta función es crucial, pues conforma el núcleo de algoritmo.

Un aspecto característico de las funciones de aptitud utilizadas en la PG, que las hace diferentes a las utilizadas por otros algoritmos de la CE, reside en el hecho de que los genotipos de los individuos son programas informáticos, y, por tanto, la evaluación de la aptitud de una solución requiere la ejecución de todos y cada uno de los programas de los individuos de la población, durante uno o varios ciclos evolutivos.

Si bien es posible usar cualquier función de aptitud, es común usar una función de aptitud popularizada por John Koza [112]. Esta función de aptitud esta estandarizada de forma que el "0" representa el resultado ideal e  $\infty$  constituye el peor resultado posible. Este comportamiento se puede invertir, de manera que la mejor aptitud se corresponda con el "1", o resultado ideal, y el "0" simbolice el peor ajuste posible, usando la función mostrada en 3-21:

$$P_g = \frac{1}{1 + \text{fitnessEstandarizado}} \quad (3-21)$$

En esta tesis, se ha utilizado una función de aptitud que proporciona valores válidos tanto negativos como positivos, siendo el rango de la función de aptitud de  $-\infty$  a  $+\infty$ , valor que es directamente proporcional al de la función objetivo.

### 3.1.6 Elitismo

El elitismo, cuyo término fue acuñado por De Jong [50], es una técnica explotativa que amplía a los AG al introducir directamente al individuo, o los individuos más aptos, de una generación en la siguiente. Algunos de los métodos de selección expuestos anteriormente permiten la desaparición de la población del individuo más adaptado, dependiendo de un valor de probabilidad de supervivencia definido. Aunque esto también puede ocurrir accidentalmente en los procesos naturales, en muchos de los AG es interesante que el mejor individuo de cada generación sobreviva en la siguiente, mediante alguna cláusula de longevidad establecida. El elitismo [50], en determinadas circunstancias, puede tener un alto impacto en el funcionamiento de la PG asegurando que no se pierdan soluciones parciales. Su uso moderado mejora la convergencia de los AG, y la calidad de la solución obtenida. Por contra, si su uso es excesivo puede llevar a una convergencia prematura de la población sobre un mínimo o máximo local, según sea el problema de que se trate.

El mecanismo de aplicación del elitismo es simple; permitimos que el algoritmo de la PG inyecte directamente en la siguiente población el, o los, individuos más aptos de la población anterior. Al mantener al mejor individuo (o individuos) en las poblaciones futuras, este mecanismo dota al algoritmo de propiedades de explotación similares a la estrategia generacional  $(\mu + \lambda)$ . Esta explotación puede producir una convergencia prematura si no se mantiene bajo control el proceso por algún medio; ya sea aumentando el ruido en la mutación y en el cruce; debilitando la presión de selección; o reduciendo el número de élites que se progresan entre las sucesivas generaciones. En este trabajo se aplicará el elitismo de un único individuo élite.

### 3.1.7 Control de tipos

El conjunto de funciones y terminales, que son permisibles, constituyen el conjunto de primitivas, que forman parte importante del proceso de búsqueda

en el espacio de soluciones de un sistema de PG [159]. Babovic and Keijzer [13] señalaron que el conjunto de primitivas de un programa de PG debe exhibir dos propiedades fundamentales: una propiedad de “cierre” (closure en inglés) y una propiedad de “suficiencia”.

La propiedad de cierre<sup>15</sup>, a su vez se puede dividir en las propiedades de consistencia de tipos y de evaluación de seguridad. La consistencia de tipos es necesaria, porque en el cruce de tipo subárbol se pueden mezclar y enlazar nodos arbitrariamente durante el proceso evolutivo. Como resultado de este mecanismo, es imprescindible que se pueda usar cualquier subárbol en cualquiera de las posiciones que forman los argumentos de cada una de las funciones del conjunto de funciones, ya que siempre es posible que un cruce de un subárbol genere esa combinación [116]. La evaluación de seguridad también garantiza que todas las funciones puedan aceptar todas las entradas posibles de los elementos terminales, para devolver un valor bien definido. Por ejemplo, el uso de una función de división puede presentar “entradas ilegales”, como es el valor de cero cuando aparece en el denominador. La operación de cierre ha de asegurar, que en ese caso, no se devuelve un valor indefinido que puede distorsionar el funcionamiento del algoritmo [112]. Por lo tanto, la evaluación de seguridad ha de garantizar el funcionamiento del algoritmo, permitiendo devolver un valor específico cuando se confronta con esas entradas ilegales. La propiedad de cierre se puede expresar mediante la definición 3.1.

**Definición 3.1.** *Siendo  $T$  el conjunto de terminales y  $F$  el conjunto de funciones no terminales, la propiedad de cierre se expresa como  $\forall f \in F, f$  puede tomar cualquier  $t \in T$  o  $g \in F$  como argumento.*

La propiedad de suficiencia, por otra parte, asegura que los elementos de conjunto primitivo han de ser capaces de expresar una solución dada al problema.

La elección del algoritmo y el modo de inicialización es importante ya que, para que la PG funcione de manera efectiva, se requiere que los conjuntos de funciones y terminales también cumplan con la consistencia de tipos de la propiedad de cierre. Una función formada por una expresión aritmética que devuelve un valor resultado ha de cumplir que este ha de ser siempre del mismo tipo consistente (int,

<sup>15</sup> En matemáticas una operación sobre un conjunto cumple la operación de cierre, cuando al realizar la operación sobre los miembros del conjunto siempre se produce como resultado un miembro de ese conjunto.

long, float, boolean) y todos sus argumentos lo han de ser también. Esta restricción se puede flexibilizar cuando se incluyen expresiones relacionales o booleanas mediante un mecanismo de conversión de tipos, estableciendo, por ejemplo, un único tipo de trabajo (float), en donde los números negativos representen el valor booleano falso y los positivos el verdadero. Koza [116] advierte que mecanismos de conversión como este pueden, sin embargo, introducir sesgos inesperados en el proceso de búsqueda, por lo que deben ser utilizados con cautela, y pone como ejemplo la función booleana *if* que puede ser sustituida por una función de cuatro argumentos, donde el requisito de consistencia de tipos, que parecía bastante limitante, se resuelve con la reestructuración de la función *if* para resolver el problema<sup>16</sup>.

Dado que las restricciones de consistencia son a menudo impulsadas por o expresadas mediante un sistema de tipos otra opción, para evitar este problema, diferente a la de realizar una conversión de tipos al vuelo, consiste en la incorporación de los tipos y sus restricciones en el propio sistema de PG. Con un sistema de PG fuertemente tipado bastaría con comprobar que las acciones de inicialización de la población, y de los operadores genéticos de cruce o mutación, no violen las restricciones del sistema de tipos. Por ejemplo Luke Sean [129] propone que cada nodo del árbol tenga asignado un tipo de retorno, y en el caso de los nodos no terminales que estos tengan definidos los tipos de retorno de cada sucesor, que se corresponde con uno de sus argumentos. Además, se ha de cumplir que:

- Para cada nodo padre y sus hijos en el árbol, el tipo de los hijos ha de ser compatible con el tipo definido en el padre, para esa posición del argumento ocupada por el hijo.
- El tipo de retorno del nodo raíz ha de ser compatible con el tipo definido para el propio árbol.

Un posible árbol generado, con control fuerte de tipos, tendría una estructura parecida a la de la figura 3.8.

---

<sup>16</sup> a, b, c, d 'If a < b then return value c, else return d'. En este ejemplo a y b han de ser dos elementos comparables, pero c y d pueden ser otros tipos arbitrarios devueltos por la función, no necesariamente booleanos.

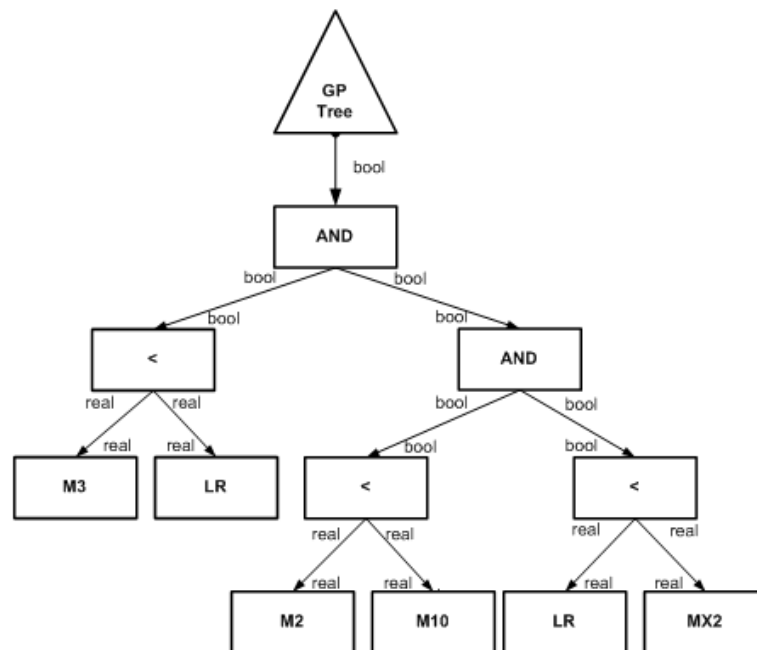


Figura 3.8: Árbol de tipos en programación genética.

En línea con lo propuesto por Sean Luke [129], en este trabajo se ha utilizado una versión mejorada de la programación genética STGP [141], que impone restricciones fuertemente tipadas de datos en el árbol. La STGP permite el cruce y la mutación de árboles sólo cuando los resultados propuestos son conformes con la restricción, de que el tipo de retorno de cada nodo ha de “encajar” con el tipo de argumento correspondiente en el padre; además, el tipo de retorno del nodo raíz debe “encajar”, a su vez, con el tipo de árbol.

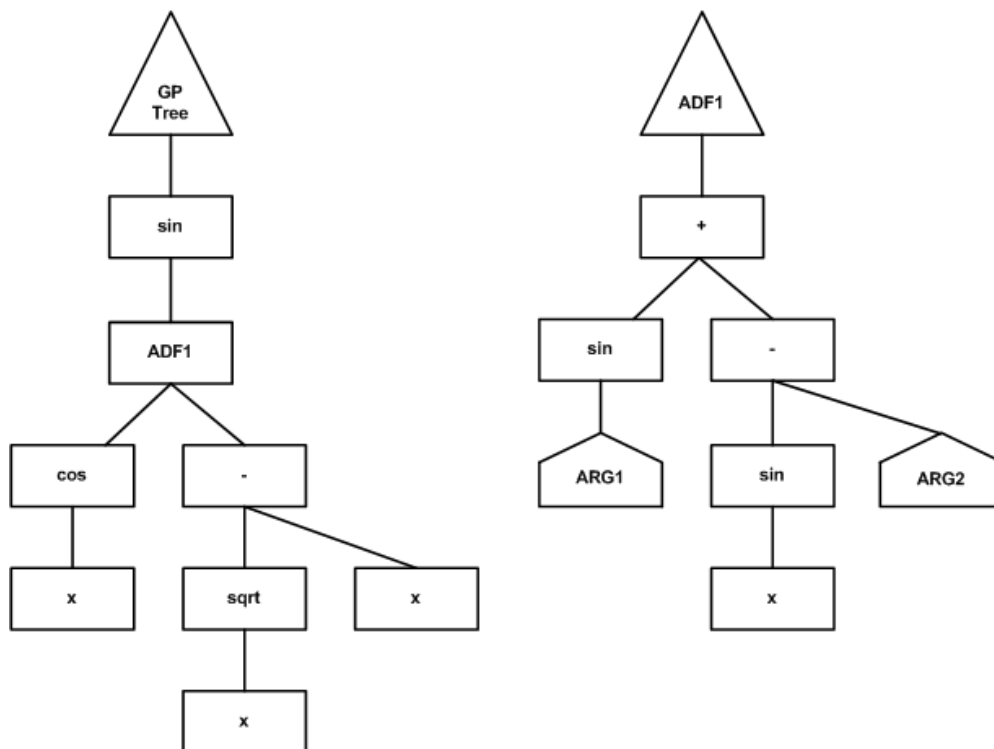
Esta solución, sin embargo, no está exenta de problemas, ya que la mayoría de los algoritmos constructores de nodos requieren que, para cada tipo accesible utilizado por la función establecida en el nodo, deba existir al menos un terminal (e idealmente un no terminal) en el conjunto de funciones que devuelva ese tipo. Esto es así porque muchos algoritmos de generación de nodos, en la PG, insisten en ser capaces de emitir un terminal cuando lo deseen, independientemente de cuál sea la situación actual del tipo esperado, donde el terminal tenga que ser colgado. Una excepción a este comportamiento es la ofrecida por el algoritmo de inicialización *Uniform*.

El problema de concordancia de tipos también se da en los operadores de cruce y mutación. En ambos casos una situación probable es que los individuos elegidos

tengan dificultades con la restricción de tipos. Por ejemplo, para establecer un cruce entre los subárboles de dos individuos, los nodos donde se colgarán deben ser compatibles con el tipo de los nodos principales del subárbol, de lo contrario las ubicaciones seleccionadas no serían válidas, o puede que no exista una posición permitida. En estos casos se reintentará la operación de elegir nodos compatibles un número determinado de veces; si este procedimiento de solución falla en todos los intentos, entonces se devuelven los padres originales en lugar de los hijos generados.

### 3.1.8 Funciones definidas automáticamente

Las funciones definidas automáticamente (ADF) [110] son una forma estándar de crear módulos de programación reusables en PG. Su composición se establece en uno o varios árboles pertenecientes a un individuo, separados del principal, que definen una estructura de funciones en la que ciertos árboles pueden llamar a otros árboles que actúan como funciones o subrutinas del llamador.



**Figura 3.9:** Ejemplo de función de PG definida automáticamente.

En el ejemplo de la figura 3.9 [129] se representa un individuo con dos ar-

gumentos. El árbol de PG principal contiene una función dentro de un nodo que llama al árbol ADF. Esta función evalúa primero a sus hijos, que son parte del árbol principal, y luego ese ejecuta el propio árbol ADF. En el árbol ADF las dos funciones terminales (ARG1 y ARG2) son las que devuelven los valores de los dos hijos ( $\cos(x)$  y  $\sqrt{x} - x$ ), cuando son evaluadas. El valor de retorno del árbol ADF ( $\text{sen}(\cos(x)) + \text{sen}(x) - (\sqrt{x} - x)$ ) se convierte en el valor devuelto por la propia función ADF al árbol principal.

En la figura 3.10 se muestra la regla de inversión de la figura 3.2 en la que se representa a esta incluyendo una ADF.

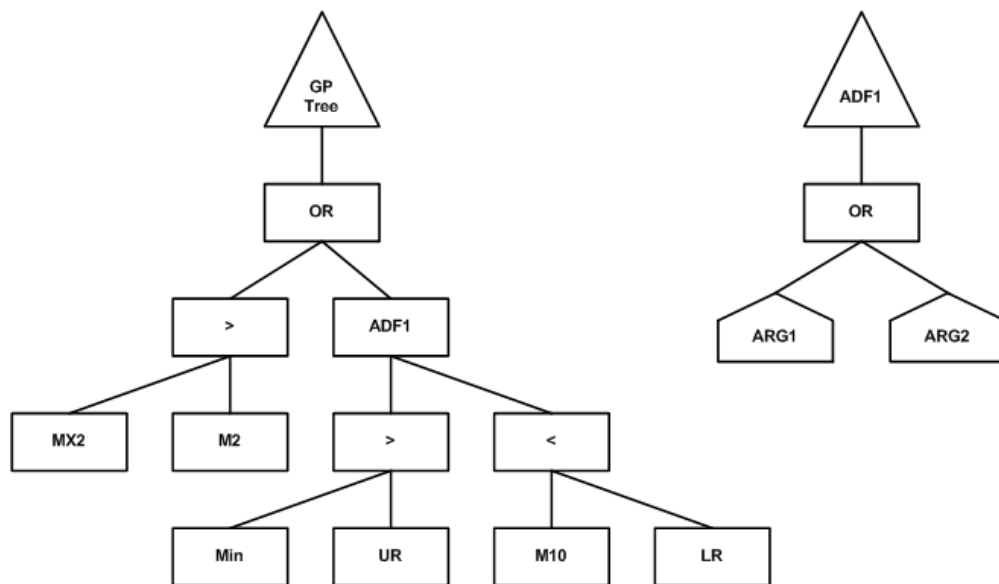


Figura 3.10: Ejemplo de regla de inversión con ADF.

Existe una variante de las ADF conocida como macros definidas automáticamente [200] (Automatically Defined Macros (ADM)). Se diferencia del comportamiento de las ADF, en cuanto a que en las ADM la evaluación de los argumentos se realiza con posterioridad a la llamada a la ADM, es decir, una vez se está ejecutando el árbol de la ADM.

### 3.1.9 Condiciones de parada

La terminación de la ejecución de cualquier algoritmo de CE, es un aspecto crucial que se ha de contemplar desde el principio, y que va acompañado de la

entrega del resultado. Los criterios de parada son variados, por ejemplo Sivanandan [196], enumeró las siguientes condiciones de finalización del proceso evolutivo del algoritmo:

- **Criterio del máximo número de ciclos evolutivos transcurridos.** El AG se detiene cuando se ha evolucionado la población un número de generaciones específico.
- **Criterio del tiempo transcurrido.** En este caso el AG se detendrá cuando haya transcurrido un tiempo máximo prefijado al inicio del proceso. Esta solución es posible combinarla con otros criterios, como el de un número de ciclos generacionales. En este caso, la condición de parada que se cumpla en primer lugar (tiempo, o número de generaciones) determinará la terminación del proceso.
- **Criterio de la aptitud del mejor individuo.** La condición de finalización está establecida por la falta de mejora en la aptitud del mejor individuo. El proceso evolutivo de los AG busca ir refinando a los individuos dentro de las sucesivas poblaciones. Si tras un número determinado de ciclos generacionales, el mejor individuo de las sucesivas poblaciones no ha experimentado una mejora en la aptitud obtenida, el proceso parará. Igual que en el caso del criterio de tiempo se puede combinar con otros criterios, de forma que el AG finalice si: se ha alcanzado el número máximo de generaciones; el tiempo otorgado a la ejecución se ha excedido; o no hay mejora de aptitud del mejor individuo. La condición que primero se cumpla será la que fije la condición de parada.
- **Criterio generacional de la función objetivo.** En este caso no es la aptitud lo que se valora, sino la mejora en el cumplimiento de la solución del problema. Aunque el criterio de aptitud del individuo, si está correctamente diseñado, debería ser directamente proporcional al de la función objetivo, esto puede que no siempre sea así. La parada se establece si no se mejora la función objetivo de un individuo en un número determinado de generaciones consecutivas.
- **Otros criterios de parada basados en la aptitud de la propia población.** Normalmente se trata de criterios estadísticos que afectan a toda la población



en su conjunto, como, por ejemplo: la media, la moda, la desviación típica de la población o su varianza. Cuando la población alcanza un determinado valor de convergencia en el indicador, o conjunto de indicadores, estadísticos utilizados, la ejecución del algoritmo finalizará.

- **Criterio basado en el mejor o el peor individuo.** Cuando en una población el valor de aptitud del mejor, o del peor individuo, ha alcanzado un determinado valor, o ha caído por debajo de un umbral determinado, el algoritmo parará.

En este trabajo se utilizará, en todos los experimentos a realizar, el criterio del máximo número de ciclos evolutivos transcurridos, que se fija en un valor de cincuenta generaciones. Se ha tomado el valor de cincuenta generaciones, como referencia, a partir del utilizado por Koza en sus experimentos [112] y en la literatura en general.

### 3.1.10 Parámetros de ejecución de un algoritmo de PG

Una característica de la PG, y habitualmente de la CE, es que está sujeta a una alta parametrización. Los operadores genéticos no sólo tienen diferentes implementaciones que se pueden seleccionar, sino que también disponen de múltiples parámetros de control que ajustan las operaciones genéticas, siguiendo alguna regla probabilística o determinista, que busca obtener una ejecución más eficiente del algoritmo principal.

Aunque hay diversos estudios que tratan de analizar la influencia que, en la búsqueda de una solución, tienen los parámetros que controlan los algoritmos en la CE, como en los trabajos de Grefenstette [80] y Eiben et al. [57], estos estudios suelen ser parciales y proceden generalmente de trabajos empíricos, y no de estudios teóricos sobre la adecuada selección de los valores de los parámetros de control de un AG.

Algunos parámetros tienen una incidencia mayor que otros en el rendimiento de los AG, y quizás el que tiene un mayor impacto sea el del tamaño de la población, Goldsberg [79]. El segundo parámetro de control, que tiene más incidencia en los resultados, es el número de ciclos evolutivos, o generaciones, que ejecuta el

algoritmo antes de alcanzar la condición de parada y de presentar la solución alcanzada. En resumen, el número de generaciones, junto con el tamaño de la población, son los dos parámetros que tienen mayor influencia en los costes computacionales y en los resultados obtenidos por un AG.

Desde el punto de vista computacional la principal limitación en el uso de los parámetros, en los algoritmos de PG, basados en el tamaño de la población y en el número de ciclos evolutivos, es el tiempo empleado en el cálculo de la función de aptitud que caracteriza a los individuos, y que se emplea en su evaluación. Por otro lado, y en general, el espacio en memoria necesario para almacenar a las soluciones candidatas presenta una restricción menor<sup>17</sup>. No obstante, las situaciones son variadas, por ejemplo, en los problemas de regresión simbólica, en donde la aptitud del individuo tiene una correlación directa con la función objetivo (no hay fase de entrenamiento), es adecuado usar el tamaño de población más grande que el sistema puede manejar. Por el contrario, en otros casos, en los que el algoritmo se ejercita con una población de entrenamiento, y la población objetivo no tiene las mismas características, un alto número de individuos, o de rotaciones, provocaría un sobreentrenamiento de la población, reduciendo la efectividad de las soluciones en su predicción.

Normalmente, el tamaño de la población más utilizado en la literatura es el de 500 individuos y el número de generaciones empleadas varía entre 10 y 50 ciclos. En general, la búsqueda más productiva se realiza en esas generaciones tempranas, y si ninguna solución se ha encontrado en ese tiempo, entonces es poco probable encontrarla en una cantidad de tiempo razonable, pues entra en juego el mecanismo de mutación que produce resultados más a largo plazo.<sup>18</sup>

Podemos estimar el número de ciclos necesarios, para conseguir una convergencia óptima, a partir de la intensidad de selección que establece un esquema de selección de individuos por torneo<sup>19</sup>, mediante la fórmula 3-22 [25].

---

<sup>17</sup> Esta afirmación, depende claramente del tamaño del individuo, de la población y de la estrategia evolutiva seguida. Si el tamaño de la población es constante no supondrá normalmente un problema, pero si este crece con cada generación la situación puede ser diferente.

<sup>18</sup> Como curiosidad Dimitri Belyaev, en sus trabajos sobre domesticación en zorros, encontró que conseguía los resultados deseados tras hibridarlos entre 20 y 25 generaciones si la característica que buscada ya estaba presente en la población inicial. En caso contrario consideraba que eran necesarios otros mecanismos que funcionasen más a largo plazo como el de la mutación [22].

<sup>19</sup> En esta fórmula no se contempla la acción de la mutación pero la tomamos como una heurística aproximada.

$$I_{(t)} = \int_{-\infty}^{\infty} tx \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} \left( \int_{-\infty}^x \frac{1}{\sqrt{2\pi}} e^{-\frac{y^2}{2}} dy \right)^{t-1} dx \quad (3-22)$$

Que analíticamente para los casos en los que el número de participantes en el torneo sea  $t = 2$  y  $t = 3$  viene dada por 3-23 y 3-24.

$$I_{(2)} = \frac{1}{\sqrt{\pi}} = 0,56 \quad (3-23)$$

$$I_{(3)} = \frac{3}{2\sqrt{\pi}} = 0,84 \quad (3-24)$$

Y que de forma aproximada se puede generalizar mediante la fórmula 3-25:

$$I_t \approx \sqrt{2\ln(t) - \ln\sqrt{4,14\ln(t)}} \quad (3-25)$$

Mühlenbein et al. [143] establecieron que el número de generaciones hasta determinar la convergencia, cuando se usa selección de truncamiento y el operador genético de cruce uniforme, viene determinado por la expresión 3-26.

$$T_c = \frac{\pi \sqrt{n}}{2 I_t} \quad (3-26)$$

A partir de esta fórmula, el tiempo de convergencia se puede obtener por un método de selección arbitrario, sustituyendo  $I$  con la intensidad de selección del método de selección correspondiente. En la selección por torneo el número de generaciones hasta determinar la convergencia vendrá dado, de esta forma, por 3-27 [25].

$$T_c \approx \frac{\pi}{2} \sqrt{\frac{n}{2\ln(t) - \ln\sqrt{4,14\ln(t)}}} \quad (3-27)$$

Y para el caso de  $t = 2$  aplicando 3-23 y 3-26 se tiene:

$$T_c = \frac{\pi \sqrt{n}}{2 \frac{1}{\sqrt{\pi}}} \quad (3-28)$$

En los experimentos, en esta tesis, se ha establecido como tamaño de las poblaciones  $n$  en 500 individuos, y una selección por torneo de dos individuos, por lo que al aplicar la función aproximada de la fórmula 3-27, obtenemos como resultado que un valor esperado máximo para obtener la convergencia es de 62,25 generaciones.

Dado que en la experimentación en los algoritmos de PG y GE se hace uso de elitismo y parsimonia, a la formula anterior se le ha corregido en un 30 % conforme a 3-29 para contabilizar esa presión adicional, con lo que el resultado anterior se ha reducido a 47,18 y que en este trabajo se ha redondeado a 50 <sup>20</sup>.

$$T_c = \frac{\pi}{2} \frac{\sqrt{n}}{\frac{1,30}{\sqrt{(\pi)}}} \quad (3-29)$$

Otros parámetros, frecuentemente empleados, son los que gobiernan la ejecución de los tres principales operadores genéticos: la mutación, la selección y el cruce. En lo que respecta a la mutación, Koza consideraba que su importancia era despreciable, y que dada una población con un número de individuos suficiente su variabilidad genética bastaba para obtener resultados mediante los operadores de selección y cruce. Sin embargo, otros autores consideran que el operador de cruce tiene un impacto limitado, pudiéndose obtener mejores resultados utilizando únicamente los operadores de mutación y selección, sobre todo cuando se trabaja a largo plazo [119].

En la inicialización, es común crear la población de forma aleatoria, usando el algoritmo *Ramped Half-and-Half*, o como en el trabajo que nos ocupa *Uniform*. La profundidad del árbol se suele establecer en un rango inicial de entre dos y seis nodos. Tanto la complejidad como la profundidad inicial máxima de los árboles dependerán de múltiples factores, como son: el tamaño del conjunto de funciones no terminales, la magnitud del conjunto de elementos terminales, y las aridades de las funciones no terminales. Sin embargo, independientemente de la importancia inicial de este parámetro, la evolución posterior desplazará rápidamente a la población original de su distribución inicial [166].

En cuanto a la reproducción y la mutación se suelen emplear valores de entre

<sup>20</sup> No obstante, se han realizado experimentos previos para comprobar la bondad del parámetro utilizado encontrándose adecuado.

un 90-99 % asignados a la creación de los individuos en la reproducción, mediante el cruce simple de subárboles de los padres, dejando el 1-10 % restante para la mutación sin cruce.

### 3.1.11 Parsimonia y crecimiento del código

Uno de los problemas de la PG, derivados de la naturaleza de representación de tamaño variable de los árboles, es que con el paso de las generaciones los individuos de la población comienzan a crecer rápidamente, en un proceso que se le conoce como *bloat*. Esta expansión, o crecimiento del código, presenta serios problemas; el primero, de naturaleza computacional, es debido a que conforme crecen los individuos estos precisan más tiempo para ser evaluados; el segundo analítico, que hace que los individuos sean más difíciles de interpretar; el tercero de resultados, que ofrecen una pobre generalización y en algunos casos suelen quedar atrapados en máximos locales, o estar muy apartados del óptimo.

Este fenómeno ha sido ampliamente estudiado. Soule, Foster et al. [199] observaron que el código de naturaleza inactivo, que se localizaba en un árbol de PG, tendía a ubicarse en las partes más bajas del árbol. A partir de ahí, cuando en el proceso de reproducción se realizaba un cruce, en el que uno de los padres proporcionaba un subárbol con código inerte, la consecuencia era que la descendencia producida tenía la misma aptitud que sus padres, pero con el efecto negativo de que los hijos tenían un mayor tamaño que el de sus progenitores, y por tanto un crecimiento no deseado en el tamaño del programa solución. Este efecto observado se multiplicaba cuando los padres mostraban una alta aptitud pues, al no ser eliminados en los procesos de selección, continuaban su crecimiento en generaciones sucesivas. A los subárboles generados que no cumplían ninguna función aparente se les denominó "*intrones*", y se les asignó la bondad evolutiva de proporcionar a los individuos resistencia a ciertas operaciones genéticas como el cruzamiento u la mutación. Esta explicación no es compartida por otros autores [126], o hay otras teorías alternativas que lo explican [114], [137].

Entre los que defienden la función positiva de los intrones podemos citar los trabajos sobre compresión y crecimiento de código de Nordin y Banzhaf [150,151]. Nordin y Banzhaf entendían que aquellos individuos que presentaban un menor ratio entre el código activo y el código total de su programa, tenían una mayor po-

sibilidad de obtener un alto valor de aptitud efectivo<sup>21</sup>, entendiendo como tal una mayor probabilidad de reproducción correcta de su programa, o traspaso genético de su característica positiva a su descendencia. En el extremo, si el código activo es muy pequeño y el no activo es muy grande, las operaciones de cruce o mutación tenderán a tener lugar en puntos del código inactivo que no afectarán a la función de aptitud de un individuo.

Independientemente de la razón por la que se produce el crecimiento del código en la PG, la forma más simple y común de mantener los árboles con un bajo tamaño de complejidad, es establecer que sea ilegal producir un árbol más grande que uno de cierta profundidad. Estipulando, por ejemplo, que los operadores de cruce y mutación no puedan producir ningún hijo que tenga más de un número determinado de nodos. En general existen diversas técnicas de simplificación para controlar este fenómeno:

- Limitar el tamaño de los individuos en la inicialización y en las operaciones genéticas posteriores de cruce y mutación, es decir siempre que los individuos sean modificados.
- Eliminar los *intrones* de los individuos, mediante un operador genético adicional de simplificación que localice y elimine el código inerte.
- Penalizar a los individuos que son más complejos, en favor de otros más simples.

La PG tradicionalmente soluciona este problema limitando la complejidad de los individuos, bien mediante el establecimiento de un límite en la máxima profundidad que estos podían alcanzar, o limitando el máximo número de nodos que podían contener. Esta solución simplista ha ido evolucionando hacia otras concepciones más modernas, que en vez de limitar a los individuos en su forma cromosómica, lo que hacen es introducir un elemento de castigo que penaliza a los individuos más complejos. Esto se suele llevar a cabo actuando sobre la función de aptitud con el objeto de que, al penalizar a los individuos dependiendo de su valor de aptitud, lo que se reduzca sea la posibilidad de que sean seleccionados

---

<sup>21</sup> El valor de aptitud efectivo, presente en la teoría de bloques, no se debe confundir con el valor de la función de aptitud utilizado en la evaluación del individuo.

en la reproducción, en vez de considerarlos como individuos fallidos una vez efectuada. A esto se le conoce como presión de parsimonia o *Parsimony*. La presión de parsimonia se puede establecer de dos maneras diferentes [131]:

- Presión de *parsimonia lineal*. Esta tiene en cuenta el valor de aptitud y complejidad de los individuos. La complejidad es introducida en el cálculo de la aptitud del individuo con un efecto penalizador. El problema de esta aproximación es que se ha de conocer la proporción correcta entre el valor de aptitud y la complejidad para establecer una fórmula penalizadora que sea adecuada.
- Presión de *parsimonia* no paramétrica. En este caso, no se tienen en cuenta los valores de aptitud y complejidad de los individuos, sino que se elige a uno entre varios en competición, en función de quien de ellos sea más apto y menos complejo. Un ejemplo de este tipo de presión de *parsimonia* es la *parsimonia lexicográfica* [123,180]. Esta se introduce en el operador de selección para que en caso de que, por ejemplo, haya empate en el valor de aptitud de los individuos involucrados en un torneo, gane aquel que presente una menor complejidad.

En este trabajo se hace uso de la presión de *parsimonia lexicográfica*, a la que se combina con un sistema de torneo de dos individuos.

## 3.2 Gramáticas evolutivas

### 3.2.1 Introducción

Esta técnica de la CE, introducida por Ryan et al. [153], está estrechamente relacionada con la PG, y aunque el objetivo y el bucle principal de ambos algoritmos son los mismos, hay diferencias tanto en la representación de los individuos, como en la forma en que se publican y actualizan las soluciones. Las GE proporcionan ventajas en los siguientes aspectos: el control de tipos, que es más flexible; la simplicidad de las soluciones obtenidas; la flexibilidad en la definición de los genomas y la clase de problemas que solucionan.

### 3.2.2 Representación

Tanto la PG, como las GE, parten del mismo concepto de recombinación genética de la información de los padres, y de los procesos de mutación que afectan a los hijos; pero mientras las GE lo que recombinan son cadenas de estructuras similares a "ADN", que son interpretadas por las gramáticas para producir las soluciones o programas, lo que la PG recombina son partes del árbol, a las que podemos considerar como propias "proteínas" finales o trozos de programa.

En la PG, los individuos suelen ser programas, en forma de expresiones-s de LISP, representados como árboles sintácticos. Los operadores genéticos actúan directamente sobre el árbol que representa al individuo y aunque, a primera vista, esto puede parecer ventajoso, limita enormemente esta técnica. La dificultad estriba en que la PG carece de cromosomas [65], siendo sus árboles estructuras mucho más complejas sobre las que actúan los operadores. Los individuos en la PG funcionan simultáneamente como genotipo y fenotipo sin que haya una clara diferenciación entre ambos. En las GE, por otro lado, si hay una clara diferencia entre estos dos aspectos que definen a un individuo. Los genotipos de las GE son vectores de enteros, que codifican la selección de reglas de una gramática libre de contexto predefinida, y a los que se aplican los operadores genéticos de los AG tradicionales, como la mutación y el cruce. El fenotipo, sin embargo, suele estar representado por una traducción previa a una estructura de árbol que después se evalúa recursivamente para obtener la solución, o calcular la aptitud. Para evaluar la aptitud del individuo se establece una relación de traducción entre el genotipo y el fenotipo. Aquí, los árboles no están destinados a la representación genotípica del individuo, sino que funcionan como estructuras temporales utilizadas en el curso del traducción y posterior evaluación de los individuos.

Según Thorhauer y Rothlauf [207], la representación de las GE afecta al tipo de problemas que estas pueden resolver de manera eficiente, pues esta está sesgada y tiene dificultades con las estructuras de solución de algunos problemas. Según su estudio, las GE tienden a generar soluciones cuyas estructuras, en los árboles de traducción construidos, son estrechas y profundas, superando en este caso a las aportadas por la PG, cuando las soluciones óptimas al problema investigado se ajustan a esta característica. Por otro lado, Thorhauer considera que las soluciones que requieren árboles más densos son más fáciles de resolver por la PG.



Un aspecto importante de las GE, que proporciona un distinto comportamiento frente a la PG, es que una misma solución puede ser representada a partir de la definición de diversas gramáticas, y por tanto, el comportamiento de las GE ante un mismo problema puede variar dependiendo de la gramática concreta que esté siendo utilizada. Incluso es posible definir diferentes gramáticas para abordar un problema, formando diferentes poblaciones de individuos que funcionan como si de razas, especies o poblaciones de diferentes islas se tratara. Las diversas gramáticas, así definidas, pueden hibridar entre sí o no hacerlo. Esta característica confiere a la técnica de las GE una grandísima ventaja, cuando se tenga conocimiento, o intuición, de cómo sea la solución buscada, pues permite guiar al algoritmo por el espacio de soluciones del problema de forma muy eficiente.

### 3.2.3 Inicialización

Con respecto a la inicialización de la población en las GE, esta parte de una definición gramatical en formato BNF que describe la sintaxis del tipo de programa utilizado para resolver el problema, junto con la especificación del conjunto de funciones no terminales y elementos terminales que se utilizarán para este fin. Acompañando a la gramática se ha de contar con los cromosomas de los individuos, representados por medio de cadenas o vectores de enteros aleatoriamente generados, que contendrán los genotipos de los individuos a construir.

A partir de ahí, el algoritmo de construcción de los individuos, como primer paso, genera vectores aleatorios de números enteros, cada uno en el rango de "0" a un valor máximo definido, al menos, por el número de producciones de la regla gramatical con el mayor número de ellos menos uno. Cada posición del vector indica la regla de la gramática que se seleccionará, después de aplicar a su valor el módulo (operación de resto) definido por el número de valores posibles de la regla considerada.

Matemáticamente, cada regla gramatical  $R_i$  representa a sus individuos como un elemento que pertenece a un grupo finito en  $\mathbb{Z}_{n_i}$ , donde  $n_i$  es el número de producciones posibles de la regla  $i$ . Un aspecto a tener en cuenta es que el vector generado de números enteros contendrá valores que exceden el módulo y, por lo tanto, son parte de la clase de equivalencia de uno de los elementos del grupo finito. Por ejemplo:

$117 \equiv 5 \pmod{7}$  pertenece a la clase de equivalencia  $|3|$

Debido a esta característica, diferentes vectores de enteros pueden corresponder exactamente al mismo individuo, con el mismo árbol de representación correspondiente traducido.

Por lo tanto, todos estos individuos forman parte de una clase de equivalencia de un individuo  $|s|$ , que está representado por las clases de equivalencia de los módulos de las reglas de producción gramaticales asociadas, con cada posición del vector cromosoma del individuo.

Dado que la inicialización del vector se consigue mediante la concatenación de números enteros generados aleatoriamente, es importante tener en cuenta el valor máximo del número a generar, que si bien se ha establecido que ha de ser como mínimo el máximo número de producciones de la regla gramatical con el mayor número de ellos menos uno, si adoptásemos sin más este valor y teniendo en cuenta la propiedad de pertenencia a clases de equivalencia de cada uno de los elementos del vector, frente a la regla de la gramática a la que corresponde, nos encontraríamos que algunos elementos de producción de las reglas se verían probabilísticamente más favorecidos que otros. Para evitarlo, se ha de elegir un valor que sea el mínimo común múltiplo del número de producciones de todas las reglas expresadas en la gramática, eliminado así la posibilidad de que haya valores que favorezcan a determinados elementos dentro de una regla.

Tómese como ejemplo un vector de una GE, que representa a la regla de inversión vista anteriormente en la figura 3.1, y la gramática correspondiente a la misma, expresada en forma BNF de la tabla 3.1, para mostrar el proceso de inicialización.

En primer lugar, supóngase que tenemos la siguiente secuencia pseudoaleatoria de enteros, que forma el vector cromosoma de un individuo, cuyos valores se establecen en el rango que va desde el valor "0" al "255":

064 176 183 171 196 211 073 067 222 129 007 031

Para poder construir la representación en árbol asociado al individuo, o validar que se trata de un individuo sano, se tiene que enfrentar el vector de enteros

**Tabla 3.1:** Gramática usada en la regla de inversión de la figura. 3.1.

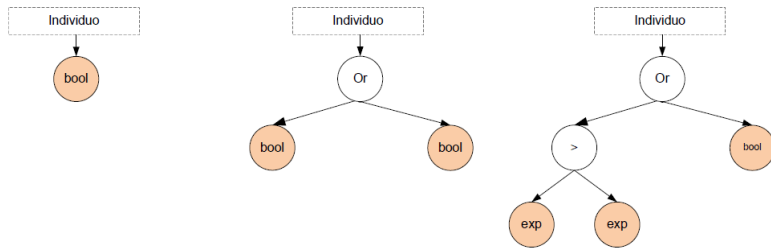
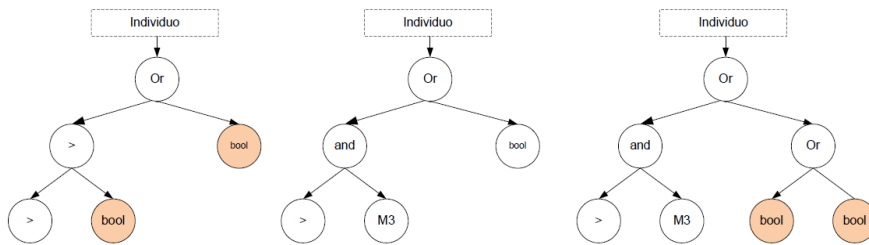
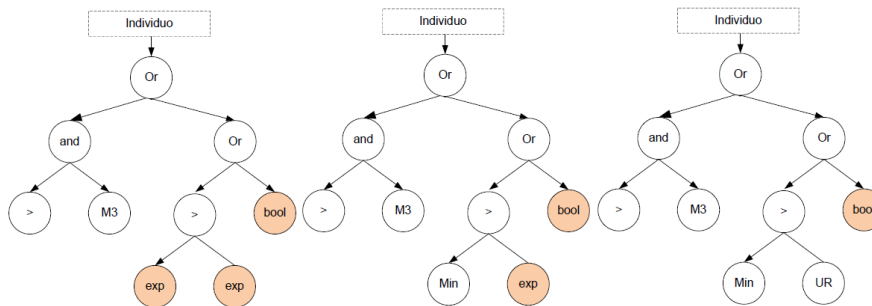
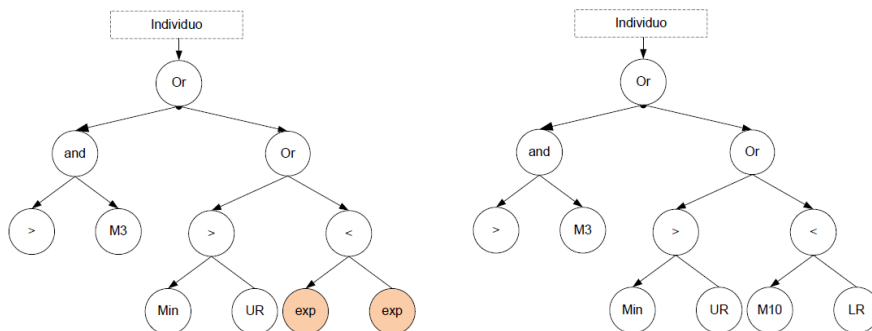
Nº	Módulo	Regla gramatical
1	1	<Rule> ::= <bool>
2	5	<Bool> ::= ( <b>And</b> <bool> <bool>)   ( <b>Or</b> <bool> <bool>) <Bool> ::= ( <b>Not</b> <bool>) <Bool> ::= (> <exp> <exp>)   (< <exp><exp>)
3	16	<Exp> ::= ( <b>Apertura</b> )   ( <b>Cierre</b> )   ( <b>Máximo</b> )   ( <b>Mínimo</b> ) <Exp> ::= ( <b>M2</b> )   ( <b>M3</b> )   ( <b>M5</b> )   ( <b>M10</b> ) <Exp> ::= ( <b>Roc3</b> )   ( <b>Roc12</b> ) <Exp> ::= ( <b>Mx1</b> )   ( <b>Mx2</b> )   ( <b>Min1</b> )   ( <b>Min2</b> ) <Exp> ::= ( <b>UR</b> )   ( <b>LR</b> )

a la gramática. Siguiendo el ejemplo, tómesese el primer número del vector de entero; 064, y calcúlese su (mód 1) asociado con la primera regla raíz de gramática, por lo que se toma la regla <bool>. Esta es una regla no terminal, por lo que se toma el siguiente número entero, 176, que en (mód 5) es 1 y, por lo tanto, se elige la segunda producción de *bool* (*Or* <bool> <bool>). *Or* es una función no terminal y se coloca en la raíz del árbol del individuo. Los hijos del nodo *Or* también deben construirse, de modo que se toma el siguiente entero 183 que en (mód 5) corresponde a 3, y se escoge la regla (><exp> <exp>). Con el siguiente entero, 171, seleccionamos la expresión (*Mx2*) un elemento terminal que se agrega al árbol. El proceso continúa construyendo el árbol en preorden, hasta que todas las reglas se expanden o el vector de enteros se agota. El proceso se muestra en forma gráfica en la figura 3.11.

El vector de enteros del ejemplo anterior es un equivalente isomorfo al individuo canónico expresado por el vector, cuyos valores numéricos de sus genes se corresponden con los números:

000 001 002 011 004 001 002 003 014 003 003 015

Fuera del contexto de evaluación de la condición de aptitud, en la que los individuos se representan en una estructura de árbol, los cromosomas de los individuos se forman por simples cadenas binarias a las que se aplican los operadores genéticos estándar de los AG tradicionales [77], como son el cruce por uno o varios

**Figura 3.11:** Construcción de un individuo en GE**(a)** Construcción de un individuo en GE paso 1.**(b)** Construcción de un individuo en GE paso 2.**(c)** Construcción de un individuo en GE paso 3.**(d)** Construcción de un individuo en GE paso 4.

puntos o la mutación, con la diferencia de que estos individuos están compuestos por genotipos que son vectores de números enteros de longitud variable. Por lo tanto, los operadores en las GE pueden implementarse de manera muy eficiente y la penalización de rendimiento al realizar la asignación también es baja. Sin embargo, las restricciones sintácticas y de tipo impuestas por las gramáticas evolutivas, logran esta eficiencia a expensas de una alta proporción de individuos fallidos. Este problema se aborda en la mutación envolviendo al cromosoma e interpretándolo de forma circular y, en el caso del operador de cruce, aplicando a este en el límite del *codón*, en lugar de en una posición arbitraria dentro del vector. La aplicación de ambos mecanismos puede reducir considerablemente la tasa de individuos no válidos [154,181].

En términos de aplicabilidad, a la generación automática de reglas de inversión, las GE y la PG serían equivalentes, ya que tienen el potencial de proporcionar las mismas soluciones. Ambas pueden generar estructuras funcionales en forma de árbol, basadas en combinaciones de bloques de construcción predefinidos.<sup>22</sup>

La inicialización de una población en las GE, depende del tipo de árbol que necesitemos construir, del número de nodos, sus aridades, la profundidad máxima del árbol, su simetría, su anchura, la complejidad etc. A partir de ahí, se ha de decidir que tipo de distribución es la más adecuada, una distribución uniforme que empiece por uno y acabe en un determinado valor, o utilizar una distribución que favorezca los valores más pequeños, como la distribución geométrica. La distribución geométrica establece un valor  $\rho$ , que determina el valor de complejidad medio seleccionado  $n$ . Cuanto mayor sea  $\rho$  mayor será  $n$ , siguiendo la ecuación 3-30:

$$E(n) = m + \frac{\rho}{1 - \rho} \quad (3-30)$$

Donde  $m$  es el mínimo valor legal y  $\rho$  la probabilidad de elegir un valor más grande.

En este trabajo la inicialización de la población, y la creación de los individuos, en las GE se define mediante una serie geométrica con un valor de complejidad mínimo inicial de cinco y una probabilidad de crecimiento de 0,85. El máximo

<sup>22</sup> Aunque en este trabajo se han utilizado operadores y funciones terminales, como bloques de construcción para construir las reglas, nada impide contar con operadores booleanos y reglas predefinidas que formen los propios bloques de construcción.

valor que pueden adoptar los bytes (genes), que constituyen el vector de enteros que representa al genoma de un individuo, viene establecido a partir del máximo común múltiplo del número de producciones gramaticales de todas las reglas, y que es 80.

### 3.2.4 Operadores

El cruce entre individuos, cuyos genomas están representados en forma de cadena (o vectores de enteros), implica que los métodos que utilizemos han de ser conscientes de que hay cierta vinculación entre los genes, o grupos de ellos, dentro de un cromosoma. Esta vinculación, o epistasis en términos de biología, que se produce entre los diferentes bloques de construcción genéticos, presenta una correlación muy fuerte con el valor de aptitud de los individuos, y determina que la PG, las GE, y en general las técnicas de CE, deberían protegerse contra la disrupción de ciertas combinaciones de genes, durante la operación de cruce.

Las técnicas de cruce habituales de los cromosomas, que están representados en forma de cadenas, son muchas, algunas de ellas son las siguientes:

- **Cruce básico. Single Point Crossover (SPX):** se selecciona un punto al azar dentro de la cadena. La parte anterior del punto elegido, a la que llamamos cabeza, es copiada del genoma del padre en el primer hijo y la posterior, o cola, del de la madre. Si hubiera un segundo hijo se conformaría mediante la cabeza de la madre y la cola del padre invirtiendo los papeles. Figura 3.12

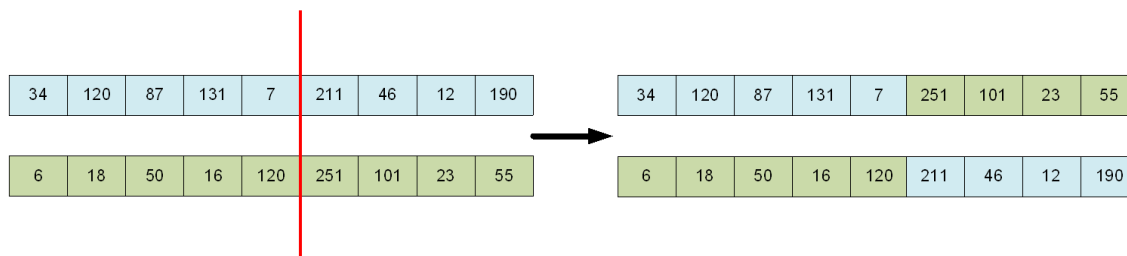
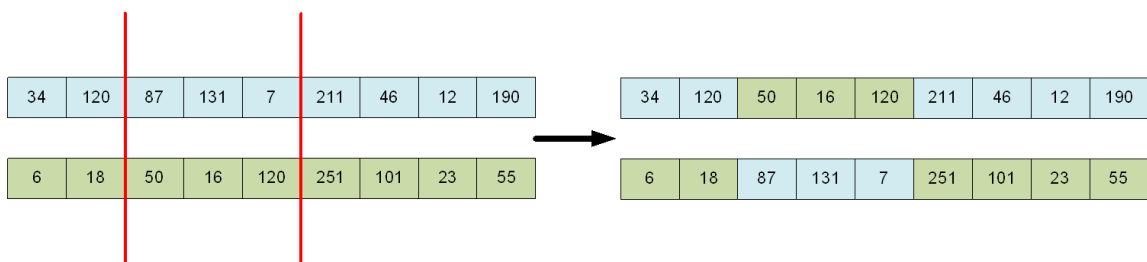


Figura 3.12: Cruce de cromosomas en las GE por un punto.

- **Cruce de dos puntos. Double Point Crossover (DPX):** su comportamiento es igual que el de cruce por un único punto, sólo que se establecen un punto adicional de cruce. Constituye una generalización del caso básico de cruce

por un punto. En vez de elegir una sola posición en el interior de los cromosomas de los padres, se escogen dos, efectuando dos cortes en vez de uno. Es interesante observar que ninguno de estos puntos de corte coincida con el límite de los cromosomas. Para obtener la descendencia se escoge el segmento central (cuerpo) de uno de los padres y los segmentos laterales (cabeza y cola) del otro progenitor. Generalizando, se pueden añadir tantas localizaciones de cruce como se desee, obteniendo, de esta forma, algoritmos de cruce multipunto. Figura 3.13.

El cruce por dos puntos mejora de forma significativa el cruce por un único punto. Sin embargo, al añadir un mayor número de puntos de cruce, se reduce el rendimiento del algoritmo genético. El problema, es que, al añadir un mayor número de puntos de cruce, se rompan los bloques de construcción genéticos, al interrumpirse las relaciones entre los genes de los segmentos originados en el cruce. Es decir, que las características de bondad, que el bloque o segmento poseía conjuntamente, se degradan o pierden efectividad. Por otra parte, también se obtienen ventajas, pues al aumentar el número de puntos se consigue explotar la variabilidad de la población, y como consecuencia que el espacio de búsqueda se explore con más profundidad [134].

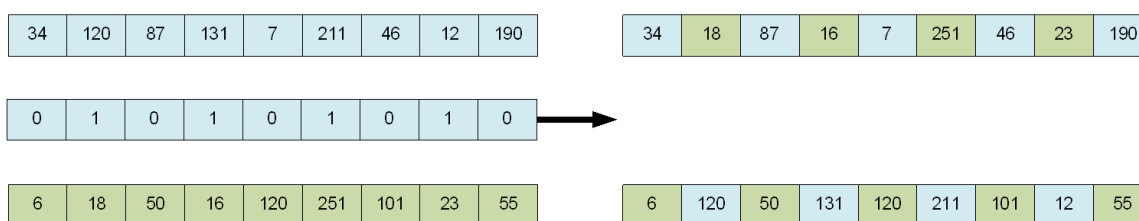


**Figura 3.13:** Cruce de cromosomas en las GE por dos puntos.

El cruce de uno o dos puntos, supone que los cromosomas están estructurados de forma que los genes que tienen relación entre sí están próximos entre ellos, siguiendo un principio de localidad, y, por tanto, la posibilidad que el grupo destruya ese grupo de construcción es pequeña. Esa suposición es muy fuerte, y su veracidad dependerá enormemente de la organización interna del cromosoma. El cruce uniforme no hace tal tipo de suposición y es adecuado para aquellos casos en los que los genes básicamente sean independientes unos de otros.

- Cruce uniforme. Uniform Point Crossover (UPX):** el cruce uniforme es la máxima expresión del cruce multipunto, estableciendo probabilidades para determinar la procedencia del valor de cada posición del ADN del hijo. En este caso, para cada uno de los genes de la cadena del descendiente, se establece una probabilidad de este pertenezca al padre, y otra de que provenga de la madre. Lo habitual, pero no tiene por qué ser así<sup>23</sup>, es que cada gen de la descendencia tenga las mismas probabilidades de pertenecer a uno u otro progenitor.

La técnica se puede implementar de muy diferentes maneras, por ejemplo, generando una máscara binaria de bits de forma aleatoria cuya longitud sea igual a la cadena que representa el genoma del individuo. Si en una de las posiciones de la máscara aparece un "1", el gen situado en ese punto, en uno de los descendientes, se copia del primer progenitor. Si por el contrario se presenta un "0" el gen se copia del segundo progenitor. Para producir el segundo descendiente se pueden seguir dos mecanismos diferentes, o bien se intercambian los papeles de los padres, o se invierte la interpretación que se hace de los unos y los ceros de la máscara de cruce. La descendencia, por tanto, contiene una mezcla de genes que proceden de cada uno de los padres. Aunque, en el operador genético, el número efectivo de puntos de cruce es fijo, se puede determinar que por término medio será de  $L/2$ , siendo  $L$  la longitud en número de genes del cromosoma. [134]. Figura 3.14.



**Figura 3.14:** Cruce uniforme de cromosomas en las GE.

Existen otras muchas técnicas de cruce como las de: cruce segmentado, cruce por permutación, cruce de orden, cruce de ciclo etc. En este trabajo se ha optado por el cruce en un único punto.

La mutación de vectores tiene infinidad de posibles variantes, pero en este trabajo sólo se tratarán sucintamente dos de ellas. Dentro del vector o cromosoma,

<sup>23</sup> Se podría dar más probabilidad al progenitor con mejor aptitud.



cada número entero individual, que pertenece al gen, puede sufrir una mutación con una determinada probabilidad. Esta mutación se puede dividir en las dos clases siguientes:

- **Mutación de bit.** En este tipo de mutación uno, o varios de los bits de un gen, adoptan un valor aleatorio de “0” o “1”. Nótese que el valor final adoptado, por el bit del gen modificado, puede ser el mismo que tenía anteriormente antes de la mutación. Una variante del procedimiento anterior, si no queremos que se repita el valor anterior del bit, consiste simplemente en cambiarlo, operación de “flip” del bit.
- **Mutación de gen.** La mutación de genes, a diferencia de la de bit, modifica completamente al gen, estableciendo en él un nuevo valor aleatorio, que se obtiene entre sus valores legales, mínimos y máximos, definidos.

En este trabajo, se ha preferido utilizar la mutación de gen, para evitar que una mutación de bit nos generase un valor que estuviese fuera de los límites deseados. La probabilidad de mutación se ha establecido en un valor común en la literatura de un 5%. Este valor, ha sido, además, confirmado tras efectuar algunas pruebas exploratorias, entre las que se barajaron valores en el rango del 1% al 10%.

Los operadores genéticos, en las GE, tienen un efecto diferente del funcionamiento asignado a la PG tradicional, que tiende a preservar la mayor parte de la estructura de los individuos y están diseñados para tener un efecto en cierto modo predecible. En las GE, por un lado, cambios menores en el genotipo pueden llevar a cambios masivos en el fenotipo por acción de la epistasis<sup>24</sup>. Por otro lado, los cambios en algunas partes del genotipo pueden no tener efecto alguno en el fenotipo. Este comportamiento tiene como consecuencia que una gran cantidad de individuos “mueran” y sean eliminados de la población, teniendo que ser sustituidos por otros nuevos.

Para compensar los problemas introducidos en el cruce y las mutaciones de los vectores, las GE cuentan con varias estrategias de reparación dentro del propio genotipo:

---

<sup>24</sup> La epistasis estudia la interacción entre diferentes genes al expresar un determinado carácter fenotípico, es decir, cuando la expresión de uno o más genes dependen de la expresión de otro gen [16].

- La primera estrategia, llamada **duplicación**, se basa en seleccionar una secuencia de enteros del propio vector, de dos índices aleatorios calculados, y agregarla al final del vector.
- En la segunda estrategia, conocida como **circular**, si al llegar al final de la cadena de enteros no se ha podido resolver la gramática, se sigue circularmente por el inicio del cromosoma, y se continúa de esa forma un número determinado de veces hasta construir el individuo, o descartarlo.
- La tercera, denominada **truncamiento**, se usa para determinar cuántos enteros del vector del cromosoma se consumen en la creación del árbol, y truncar el resto de los enteros que no se hayan utilizado.

En este trabajo de tesis, se utilizarán conjuntamente los tres mecanismos anteriores.

### 3.2.5 Control de tipos

Como ya se mencionó, en el estilo de PG convencional establecido por Koza, el conjunto de funciones debe cumplir con el requisito de cierre, por el que todas las funciones existentes deben poder aceptar como argumentos de entrada la salida de todas las demás funciones, sean estas terminales o no terminales. En la PG esto se suele lograr utilizando un sólo tipo de datos, como el uso del tipo de punto flotante de doble precisión, ajustando las funciones para realizar una coerción de tipos. Esta limitación no la sufren las GE ya que el problema es manejado por gramáticas, generalmente definidas a través de los formalismos expresados en la forma de Backus-Naur Formalism (BNF). Esta estrategia permite restringir el espacio de búsqueda, incorporando el conocimiento del dominio del problema y facilitando un control de tipos eficiente.

Por otro lado, con la PG también se sufre el problema de garantizar que los programas que surgen en la inicialización, y recombinación de los individuos, además de compatibles en tipos sean sintácticamente correctos. Con las GE este problema deja de existir, ya que al estar basados en una gramática de referencia la generación de los programas es, por definición de la misma, sintácticamente correcta.

### 3.3 Extensiones comunes a la PG y las GE

Las técnicas para la reutilización de bloques de construcción ADF son cruciales para el rendimiento de las aplicaciones que utilizan tales bloques. Es fácil trasladar las técnicas utilizadas por Koza, para la utilización de ADF en la PG, a cualquiera de los métodos basados en las GE. De hecho, es mucho más sencilla la inclusión y especificación de ADF, usando una gramática formalmente definida, que agregar las restricciones a propósito necesarias en la construcción de los operadores de la PG, basados en árboles. [149], ya que basta con definir de la misma forma gramatical a los subárboles asociados a las ADF.

Véanse los apartados *Funciones definidas automáticamente* 3.1.8 y *Parsimonia y crecimiento del código* 3.1.11.

#### 3.3.1 La búsqueda de la solución en PG y GE

Se ha definido con anterioridad que la PG tiene por objeto encontrar una solución mediante un proceso de búsqueda. La pregunta que se suscita es ¿cómo se puede caracterizar, estudiar y predecir la búsqueda de la solución en programación genética? Hay diferentes aproximaciones teóricas para describir el proceso.

La primera de estas aproximaciones está basada en el análisis de cadenas de Markov [73]. Las cadenas de Markov modelizan a los individuos dentro de la población y definen, probabilísticamente, como las operaciones genéticas los cambian a lo largo del tiempo. En estos modelos, la población se representa como un punto en un espacio multidimensional y lo que se estudia es la trayectoria de la misma hacia determinados atractores en un sistema dinámico [168].

Una segunda aproximación es la idea del paisaje evolutivo, concebida por Wright Seawall [219]. Un paisaje evolutivo es una metáfora utilizada en PG, para pensar y visualizar los procesos evolutivos, como por ejemplo la selección natural o la deriva genética, que actúan sobre una entidad biológica [220], sea esta un gen, una proteína, una población, una especie, etc. Se puede representar en dos dimensiones por medio de un eje “y”, que generalmente se atribuye al resultado de la función de aptitud, y cuya bondad se determina por el valor adoptado, y un eje “x” que determina la posición del individuo. Los pequeños movimientos a través

del espacio de soluciones causan cambios en la función de aptitud que son también proporcionalmente pequeños. Cuando las mutaciones tienen poco o ningún efecto sobre la aptitud, los paisajes se consideran suaves y estamos en la situación ideal. Por el contrario, si los pequeños movimientos producen grandes cambios en la aptitud esa condición de suavidad no se cumple.

En la representación de la metáfora, se suelen dibujar las soluciones mediante un paisaje en el que la parte más alta se corresponde con el punto solución óptimo. La tarea de encontrar la mejor solución al problema es equivalente a encontrar uno de esos puntos más altos en la montaña. Sí bien esta metáfora representa bien el uso de técnicas como *Hill Climbing*, no es adecuada para la PG, porque no se cumple con la propiedad de cercanía de la vecindad. En *Hill Climbing* cuando se efectúa una pequeña modificación, en un individuo, el proceso lleva a otro individuo cercano a nuestra propia posición, y, a base de pequeños cambios, se va desplazando poco a poco con cada modificación hacia el punto óptimo. Cuando la variable es continua es fácil determinar quiénes son los vecinos mediante su distancia de Hamming, o cualquier otro tipo de distancia utilizada (distancia Euclidiana, distancia de Mahalanobis, distancia inversa, distancia de Manhattan, etc). Sin embargo, en la PG es posible que este no sea el caso porque el operador puede crear múltiples saltos, de tal forma que los vecinos no están relacionados por su valor de aptitud, si no por dónde se puede llegar a partir del punto en el que el operador genético ha actuado. Es decir, que puntos se pueden alcanzar a partir una determinada situación genética de un individuo.

En la PG la forma de generar y mutar los individuos sufre un sesgo de codificación en los genes discretos asociados a las funciones [139], ya que un cambio en una de ellas puede desviar enormemente al individuo de su valor presente de aptitud. Por ejemplo, tomemos la función  $y = 3x^2$ , esta función nos daría  $y = 300$  para  $x = 10$ , si la función  $x^2$  fuera sustituida en una mutación por  $\log(x)$  el resultado sería  $y = 3$ , es decir, un cambio simple en el gen de una función puede generar un salto enorme en el resultado de aptitud obtenido. Otro problema es que la elección del conjunto de funciones concreto, entre las disponibles, puede dificultar, sino imposibilitar, la obtención de soluciones aceptables. Estamos, por tanto, tratando con un sistema discreto y no continuo, en el que es posible que no todos los puntos del paisaje de soluciones puedan ser representados.

Los genes que representan los coeficientes no presentan en general, como las

funciones, un sesgo de codificación, ya que un cambio pequeño en su valor provocará normalmente un cambio pequeño en el valor de la función de aptitud. Un cambio, por mínimo que sea, puede producir, sin embargo, problemas de continuidad, cuando el valor del alelo modificado haga que la función caiga fuera de su dominio creando un individuo fallido.

Lo ideal es que modificaciones pequeñas en la representación de la codificación lleven asociadas modificaciones pequeñas en el ajuste del individuo y que den como resultado soluciones similares. Esta condición de adyacencia (continuidad) no se cumple en las funciones porque el cambio en la mutación de una función a otra (por ejemplo, de seno a una logarítmica) genera una solución muy separada de la original. Este sesgo no puede evitarse y hay que tomarlo como una de las características intrínsecas del problema. Se podría minimizar al caracterizar a las funciones agrupándolas por categorías y codificándolas mediante algún sistema en el que la modificación del código favorezca a funciones similares (exponenciales, logarítmicas, trigonométricas etc.).

Otra aproximación diferente es la conocida como análisis de componentes. El análisis de componentes fija la atención en la propagación de los subcomponentes que forman los individuos. En las gramáticas evolutivas los individuos están representados por cadenas de bits y el análisis de componente considera que es lo que ocurre dentro de las cadenas, y como los bits o grupos de bits se propagan de generación en generación. La aproximación de componentes no se fija en los efectos de los individuos sino en el efecto de la evolución en los patrones de bits [49].

Quizás la aproximación más extendida es la teoría de esquemas, desarrollada por Holland [93]. Un esquema es una cadena de longitud  $l$  construida a partir del alfabeto  $A = \{0, 1, *\}$ . El “\*” representa el símbolo de “no importa”, de modo que un esquema puede corresponder a varias cadenas en el espacio de soluciones, mientras que los índices, donde el esquema tiene un “0” o “1”, se denominan posiciones fijas. El número de posiciones fijas determina el orden del esquema  $o(H)$ , y cuando una cadena coincide con el esquema en las posiciones fijas, se le denomina instancia del esquema. Las operaciones de cruce y mutación pueden interrumpir el esquema, si dan como resultado un cambio en alguna de las posiciones fijas, mientras que un cambio en uno de los otros índices resulta en un individuo diferente pero que forma una instancia del mismo esquema.

La teoría de esquemas estudia cómo estos se forman en la población y cómo se

relacionan con el proceso evolutivo, por el cual las soluciones con un mayor valor de aptitud se desarrollan paulatinamente con cada nueva generación. Dentro de la teoría de esquemas se desarrolla el concepto de bloque de construcción, que consiste en un esquema reducido con un valor de aptitud superior a la media. Los buenos algoritmos, dentro de la AG, combinan dichos bloques para ir construyendo sucesivamente mejores soluciones, e incluso se espera que dada cualquier partición de esquema, un AG converja a la clase con la mejor aptitud de media. Muchos autores critican esta hipótesis pues consideran que la evolución es un proceso demasiado aleatorio para que una afirmación como esta sea posible, y de hecho estudios empíricos han demostrado que la hipótesis de bloques de construcción es poco probable que sea cierta [10, 34, 157, 205].

Existen diferentes teorías sobre esquemas y su estudio sobrepasa el alcance de esta tesis, sólo se presentarán algunas. Estas dependen de la representación de los individuos de la población a las que están dirigidas, por ejemplo: a representaciones en forma de cadenas binarias como en los AG o a representaciones en árbol como en la PG, aunque existen algunas que son aplicables a ambas representaciones [115]. Las GE se encontrarían, en principio, en un punto intermedio por su representación dual producto de la operación de mapeo, aunque su representación genotípica en forma de vector de enteros la confiere un carácter especial debido a que sus operaciones genéticas de mutación y cruce no se corresponden exactamente ni con las de la PG tradicional, ni con las de la AG que trabajan a nivel de bit.

El teorema del esquema de Holland o teorema fundamental de los AG es quizás el más famoso de los teoremas de esquemas [93], y establece que los esquemas cortos, de orden bajo, con un aptitud superior a la media aumentan exponencialmente en cada generación. Esta afirmación viene expresada por la inecuación 3-31.

Donde:

$m(H, t)$  es el número de instancias del esquema  $H$  existentes en la población en la generación  $t$ .

$p_c$  y  $p_m$  son las probabilidades de cruce y mutación.

$f(H, t)$  es la aptitud promedio del esquema  $H$  observada en la generación  $t$ .

$\bar{f}(t)$  es la aptitud promedio de la población observada en la generación  $t$ .

$d(H)$  es la distancia definida entre la primera y la última posición, específica del esquema, en la cadena.

$l$  es la longitud de la cadena.

La expresión  $\frac{d(H)}{l-1}$  representa la fragilidad del esquema, es decir, la proporción de lugares donde el esquema puede ser interrumpido.

$o_H$  se define como el número de posiciones fijas en el esquema.

$$E(m(H, t + 1)) \geq \frac{f(H, t)m(H, t)}{\bar{f}(t)} \left(1 - p_c \frac{d(H)}{l-1}\right) o(H)p_m \quad (3-31)$$

La consecuencia de la expresión anterior es que los esquemas cuya distancia definida es más pequeña tiene más posibilidades de no ser destruidos. Por tanto, el rendimiento del AG tiene lugar a través de la unión de esquemas cortos, de bajo orden y de alto rendimiento, llamados bloques de construcción. Holland suponía que el cruce se producía de forma uniforme en un único punto y que con el paso de las generaciones las operaciones de cruce y mutación dejaban de interrumpir el esquema, ya que, al crecer el número de individuos con un mejor valor de aptitud de promedio, estos tienden a dominar la solución ignorándose así el cruce y la mutación. El teorema de esquemas de Holland identifica los bloques de construcción, de una buena solución, pero sólo se fija en los efectos destructivos de la operación de cruce, y no en los constructivos que se supone que son una gran parte del éxito de los algoritmos genéticos.

Ya en el ámbito de la PG el teorema de esquemas, Koza definió un esquema como *un subespacio de árboles que contiene un conjunto predefinido de subárboles*, con esta definición un esquema puede aparecer en multitud de lugares dentro del mismo programa. Los trabajos sobre esquemas de Koza fueron más tarde formalizados y refinados por O'Reilly y Oppacher [157], que derivaron la teoría de programación de esquemas de programación genética, en presencia de los operadores genéticos de selección de aptitud proporcional y de cruce. El teorema está basado en la idea de definir un esquema como un conjunto desordenado de subárboles y fragmentos de árbol. Los fragmentos de árbol, son árboles que al menos contienen una hoja con un símbolo #, "no importa", que puede ser emparejado por cualquier subárbol,

incluidos árboles de tan sólo un nodo. La definición de O'Reilly y Oppacher permite definir el concepto de orden y longitud de un esquema. El orden del esquema es el número de nodos "no importa" que contienen las expresiones y fragmentos de árbol del esquema, y la longitud se define como el número de enlaces que conectan a las expresiones y fragmentos de árbol, así como los enlaces que conectan a estos. La expresión de O'Reilly viene definida por la ecuación 3-32.

$$E(m(H, t + 1)) \geq \frac{f(H, t)m(H, t)}{\bar{f}(t)} \left( 1 - p_c \max_{h \in Pop(t)} P_d(H, h, t) \right) \quad (3-32)$$

Donde:

$P_d(H, h, t)$  es la probabilidad de interrupción debida al cruce del esquema  $H$ , en el programa  $h$ , en la generación  $t$ .

O'Reilly y Oppacher establecieron, en la hipótesis de los bloques de construcción en programación genética, que se combinan los bloques orden de bajo de generaciones previas, y se compactan en soluciones parciales de alta aptitud, para componer individuos que de generación en generación mejoran su valor de aptitud.

Por último citar a Poli y Langdon [167], que definieron un esquema de forma similar, 3-33, pero usando información posicional para evitar el problema de que múltiples instancias de un esquema aparecieran dentro de un mismo individuo. Un esquema de programación genética, así definido, es un árbol con raíz compuesto de nodos dentro del conjunto  $F \cup T \cup \{=\}$ . donde  $F$  y  $T$  son conjuntos de funciones no terminales y terminales, y  $=$  es el símbolo "no importa" que puede ser de la aridad que sea necesaria. Con esta definición un esquema parte el espacio de programas en subespacios de programas de tamaño fijo. De esta forma el efecto de los operadores genéticos, dentro de un esquema, son mucho más fáciles de evaluar.

$$E(m(H, t + 1)) \geq \frac{f(H, t)m(H, t)}{\bar{f}(t)} (1 - p_m)^{o(H)} \times \left( 1 - p_c \left( p_d(t) \left( 1 - \frac{m(G(H), t)f(G(H), t)}{\bar{n}f(t)} \right) + \frac{d(H)m(G(H), t)f(G(H), t) - m(H, t)f(H, t)}{(N(H) - 1)\bar{n}f(t)} \right) \right) \quad (3-33)$$



Se realizaron extensos experimentos en un programa genético para registrar todos los esquemas de la población, sus valores medios, el estado promedio de la población, el número de programas que muestrean un esquema, la longitud, el orden, la definición de la longitud del esquema, y el esquema de diversidad. Los resultados confirmaron varias conjeturas:

- Primero, la interrupción del esquema es frecuente al inicio de la ejecución del algoritmo, y antes de que la operación de cruce contrarreste los efectos de la selección. Si sólo la selección está actuando, entonces el esquema con valores de aptitud que estén por debajo de la media desaparecerá.
- En segundo lugar, sin mutación la población convergerá rápidamente y los esquemas cortos con aptitud por encima del promedio tendrán una probabilidad de interrupción más baja.
- En tercer lugar, la desviación promedio en la aptitud de los esquemas de orden superior es mayor que la de esquemas de orden bajo. Sorprendentemente, este experimento concluye que los bloques de construcción no crecen exponencialmente.

Poli y Langdon encontraron que inicialmente, el cruce de árbol por un punto era tan disruptivo como en el cruce de la programación genética estándar, pero a medida que avanzaba la ejecución del algoritmo se volvía mucho menos agresivo. En este trabajo se usa el cruce uniforme por un único punto.

*Esta página se ha dejado en blanco intencionadamente.*

## Capítulo 4

# Aspectos generales de la solución

Como ya se mencionó en la introducción, el objetivo final a alcanzar en esta tesis es construir algoritmos de computación evolutiva, que realicen una búsqueda automática de reglas de inversión, analizando para ello series temporales financieras. A partir de este análisis, se extraerán de estas series las características que definan las reglas obtenidas como: rentables, eficientes, simples, que sean fáciles de interpretar, y que además sean robustas a los cambios estructurales propios de las series financieras.

Para ello se ha realizado una evaluación comparativa de los algoritmos de CE, GE y PG, así como del impacto que sobre los resultados tiene el agregar a los algoritmos centrales técnicas de ADF y mecanismos de control de complejidad de las soluciones. Además, se estudiará la sensibilidad de las diferentes configuraciones utilizadas, al tamaño de la población y al de funciones técnicas disponibles dentro del conjunto de terminales.<sup>1</sup>

Una vez seleccionada la técnica apropiada, investigaremos soluciones basadas en sistemas de inversión dinámicos cuyos enfoques combinen, por un lado, la posibilidad de cambiar el modelo de inversión activo como una reacción a los cambios en el mecanismo de generación de precios del mercado, y por otro que cuenten con componentes de resistencia al cambio, cuyo objeto es mitigar las consecuencias de un volumen de órdenes de compra y venta excesivo.

---

<sup>1</sup> Las funciones de análisis técnico, se calculan al inicio del algoritmo quedando sus resultados almacenados como valores constantes asociados a los valores de sesión. Su comportamiento es parecido al que tendría una variable en esa posición o de funciones de aridad uno en el número de sus argumentos.

En esta sección se describe, por tanto, la configuración experimental general utilizada, que se compone de: el entorno de trabajo usado; la representación de las reglas de inversión; los conjuntos de terminales y no terminales empleados; la gramática utilizada; la función de aptitud de los individuos; las fuentes de datos utilizadas; y otros aspectos relacionados con la parametrización común de los algoritmos de la PG y de las GE.

## 4.1 Entorno de trabajo

Para la realización de la experimentación, se ha utilizado el marco de trabajo de computación evolutiva *ECJ* (Java Based Evolutionary Computation) [55,210], al que se ha complementado con el lenguaje de programación de propósito general orientado a objetos *Java* [103], los entornos integrado de desarrollo a la programación (en inglés: Integrated Development Environment (IDE)) *Eclipse* [56] y *Netbeans* [148], un lenguaje de programación con un enfoque dirigido al análisis estadístico conocido como *R* [171], y el entorno de visualización de datos basado en librerías de *Java Processing* [70].

*ECJ* es un sistema de investigación de computación evolutiva de código abierto escrito en *Java* con más de 25 versiones<sup>2</sup>. Es un marco de trabajo abierto que soporta una gran variedad de técnicas de computación evolutiva, tales como los AG, la PG, las GE, la coevolución, la optimización de enjambre de partículas (Particle Swarm Optimization (PSO) [104,192] y la evolución diferencial entre otras. El marco modela los procesos evolutivos iterativos, como los de la PG y las GE, usando una serie de tuberías (pipelines) dispuestas para conectar una o más subpoblaciones de individuos a las que se aplican diversos operadores. *ECJ* ha sido complementado con una herramienta de control y visualización en *Java* mediante la API (en inglés: Application Program Interface) de visualización de datos *Processing*, para permitir el análisis de los resultados y automatizar tanto las pruebas, como la obtención de las estadísticas. Todas las herramientas se han integrado mediante dos entornos IDE de programación, para facilitar el desarrollo de la experimentación: *Netbeans* y *Eclipse*.

---

<sup>2</sup> La versión 26 está disponible desde el 5 de Julio del 2018 en la página <https://cs.gmu.edu/~simseclab/projects/ecj/>. High-Performance Community uristics Library for Stochastic Optimization Research. Tiene asegurada su continuidad, al menos, hasta el 2021.

Los resultados obtenidos por la herramienta de control de la experimentación han sido posprocesados, por un programa desarrollado en lenguaje *R* [35,100], para obtener las estadísticas (media, moda, varianza, máximo y mínimo) y los contrastes estadísticos basados en medias o medianas.

El hardware utilizado en las simulaciones, para la evaluación de la técnica evolutiva más apropiada en la resolución del problema, ha estado compuesto por dos portátiles *I7* de cuatro núcleos con 16 gigabytes de memoria, y que han precisado entre 1.000-1.200 horas de funcionamiento para generar los cerca de 260.000 ficheros de *logs* de salida.

En cuanto a la experimentación, relacionada con la búsqueda de reglas de inversión resistentes a cambios estructurales propios de las series financieras, hemos utilizado para su realización también el marco de trabajo *ECJ* y el lenguaje de programación *R*.

Por otra parte el hardware utilizado en estas simulaciones, consistió principalmente en un servidor *ServerMicro*, de cuatro núcleos, *Intel (R) Xeon (R) CPU E3-1241 v3 3.50GHz, 3.501 Mhz* con 32 gigabytes de memoria *DDR4* y dos discos SSD de 450.000 operaciones de entrada/salida (Input Output Operations Per Second (IOPS)).

La documentación generada ha sido elaborada inicialmente en *Microsoft Word 365*, *Microsoft Visio Profesional 2007*, *Excel 365* y *Graphviz 2.38* [60], pero después portada a *Latex* (*sharelatex* [189] y posteriormente migrada a una cuenta de *overleaf* [158]).

## 4.2 Representación de las reglas de inversión

Las reglas de inversión, generadas por los algoritmos de CE, proporcionan básicamente recomendaciones binarias que indican entrar o salir del mercado. Como se discutirá más adelante, estas recomendaciones sugerirán estar en el mercado y obtener un rendimiento asociado a la revalorización del activo, o permanecer fuera del mercado y obtener un rendimiento sin riesgos. Sólo se permitirán posiciones largas, no trabajándose posiciones intradía.

La generación de las estructuras, en el núcleo de la codificación específica utilizada por la PG y las GE, requiere una discusión detallada de dos componen-

tes: los elementos básicos de construcción de los programas solución y las reglas de combinación de dichos elementos. El primero requiere describir los conjuntos de funciones no terminales y elementos terminales utilizados, mientras que el segundo implica la introducción de la gramática del lenguaje que se utilizará como referencia.

En una etapa posterior, los individuos se codificarán representados como árboles sintácticos en la PG o como cadenas de enteros en las GE. A pesar de las diferencias que presentan las dos técnicas de CE, ambas comparten los rasgos que se analizan en las siguientes secciones de este capítulo.

#### 4.2.1 Conjunto de funciones no terminales y elementos terminales

Como ya se ha dicho, las reglas de inversión se codificarán de dos formas diferentes: como árboles sintácticos en la PG y como vectores de enteros en las GE. En este trabajo, tanto los elementos terminales como las funciones no terminales reproducen a las utilizadas por Lohpetch et al. [120] en sus trabajos. Sin embargo, una diferencia que se introduce, frente al estudio de Lohpetch, es que en esta tesis se trabaja con datos bursátiles de entrada diarios en lugar de mensuales. Los elementos terminales incluyen a los indicadores técnicos y precios de sesiones pasadas; mientras que los no terminales están formados por operadores lógicos y relacionales, que proporcionan la capacidad de generar todas las posibles combinaciones lógicas que trabajan sobre los indicadores técnicos y precios de sesiones previas: (*and*, *or*, *not*, *>* y *<*).

Los indicadores técnicos utilizados en este trabajo, como elementos terminales del árbol sintáctico en la PG, o como elementos de la gramática en las GE, asociados a los individuos, son los siguientes:

- **Apertura, cierre, máximo, mínimo.** Constituyen los precios diarios del índice bursátil considerado. (*Opening*, *Closing*, *Max* y *Minimum*).
- **Índices móviles de 2, 3, 5 y 10 meses** (un mes se considera que tiene 22 sesiones). Forman los promedios móviles utilizados. Se calculan como las medias aritméticas simples de los precios de cierre, durante un número predefinido de períodos de tiempo. Son indicadores ampliamente utilizados en el análisis

técnico para detectar tendencias y suavizar las series de precios ( $M2$ ,  $M3$ ,  $M5$  y  $M10$ ).

- **Indicador de tasa de cambio (ROC)** (3 meses y 12 meses). Constituye la tasa de cambio del índice bursátil. Es un indicador técnico de impulso o velocidad, que mide la variación porcentual del precio, calculado como una relación entre el precio actual y el precio de  $n$  períodos anteriores ( $Roc3$  y  $Roc12$ ).

$$ROC = \left( \frac{\text{PrecioCierreDia} - \text{PrecioCierreDiaHaceNmeses}}{\text{PrecioCierreDiaHaceNmeses}} * 100 \right) \quad (4-1)$$

- **Indicadores de resistencia de precio (PRI)**. Son puntos de precio en el mercado que se espera que sean difíciles de romper. Consideramos los siguientes: los dos mínimos de media móvil de tres meses anteriores que constituyen los indicadores de resistencia inferior, y los dos máximos de media móvil de tres meses anteriores que se establecen como los indicadores de resistencia superior ( $Mx1$ ,  $Mx2$ ,  $Min1$  y  $Min2$ ).
- **Indicadores de línea de tendencia (TLI)**. Establecen una línea de resistencia inferior basada en la pendiente de los dos mínimos anteriores, más una línea de resistencia superior basada en la pendiente de los dos máximos previos. Las líneas de tendencia se utilizan para mostrar la dirección y la velocidad de los cambios de precio ( $UR$  y  $LR$ ).

$$m = \left( \frac{MX2y - MX1y}{MX2x - MX1x} \right) \quad (4-2)$$

$$URy = (MX1y + m(URx - MX2x)) \quad (4-3)$$

Donde  $m$  es la pendiente de la recta,  $URy$  es el valor extrapolado y  $URx$  es el valor temporal que se quiere obtener. De igual forma se calcularían  $LRx$  y

LRy a partir de *Min1* y *Min2*.

#### 4.2.2 Gramática

Como se pretende utilizar la PG con un fuerte control de tipos (STGP), las reglas diseñadas para combinar los elementos de los conjuntos de funciones, entre ellos y con los conjuntos de terminales, se aplicarán tanto a la PG como a las GE por igual, aunque la forma de expresar las restricciones de tipos en una u otra técnica, así como la complejidad de la implementación de las restricciones aplicadas, son significativamente diferentes.

La gramática creada para describir las reglas de inversión se basa en una única expresión que combina operadores relacionales y lógicos, comunes a todos los lenguajes tradicionales. Si la expresión se evalúa como verdadera se interpretará como una señal que recomienda comprar, de lo contrario devolverá falso, lo que representa una indicación, o sugerencia, para vender. La gramática construida sigue la aproximación de Lohpetch et al. [120] que se puede expresar en la forma de notación BNF como se expone en la tabla 4.1.

**Tabla 4.1:** Gramática de una regla de inversión.

Nº	Módulo	Regla gramatical
1	1	<Rule> ::= <bool>
2	5	<Bool> ::= ( <b>And</b> <bool> <bool>)  ( <b>Or</b> <bool> <bool>) <Bool> ::= ( <b>Not</b> <bool>)  (> <exp> <exp>)   (< <exp><exp>)
3	16	<Exp> ::= ( <b>Opening</b> ) ( <b>Closing</b> )   ( <b>Max</b> )   ( <b>Minimum</b> ) <Exp> ::= ( <b>M2</b> )   ( <b>M3</b> )   ( <b>M5</b> )   ( <b>M10</b> )  ( <b>UR</b> )   ( <b>LR</b> ) <Exp> ::= ( <b>Roc3</b> )  ( <b>Roc12</b> )   ( <b>Mx1</b> )   ( <b>Mx2</b> )   ( <b>Min1</b> )   ( <b>Min2</b> )

Algunas configuraciones, que extienden la técnica base de computación evolutiva, requieren el uso de funciones definidas automáticamente. En esas circunstancias, se amplía la gramática de la tabla 4.1 con la definición gramatical adicional de la tabla 4.2.



**Tabla 4.2:** Ampliación de una gramática con la definición de una ADF.

Nº	Módulo	Regla gramatical
4	1	$\langle \text{bool} \rangle ::= (\mathbf{ADF1} \langle \text{bool} \rangle \langle \text{bool} \rangle)$

Y agregamos el componente de la tabla 4.3, requerido para expresar en la gramática a la propia ADF.

**Tabla 4.3:** Definición gramatical de una ADF.

Nº	Módulo	Regla gramatical
1	1	$\langle \text{Rule} \rangle ::= \langle \text{bool} \rangle$
2	5	$\langle \text{Bool} \rangle ::= (\mathbf{And} \langle \text{bool} \rangle \langle \text{bool} \rangle) \mid (\mathbf{Or} \langle \text{bool} \rangle \langle \text{bool} \rangle)$ $\langle \text{Bool} \rangle ::= (\mathbf{Not} \langle \text{bool} \rangle) \mid (\mathbf{>} \langle \text{exp} \rangle \langle \text{exp} \rangle) \mid (\mathbf{<} \langle \text{exp} \rangle \langle \text{exp} \rangle)$
3	2	$\langle \text{Exp} \rangle ::= (\mathbf{ARG0}) \mid (\mathbf{ARG1})$

### 4.3 Función de aptitud

La definición de la función de aptitud es de suma importancia para los algoritmos de CE pues, por un lado, guía al algoritmo utilizado en su búsqueda de un individuo solución óptimo, y por otro es el elemento computacionalmente más costoso de la técnica. En este trabajo, se define la idoneidad de las estrategias de inversión simplemente como la suma de todos los beneficios de transacción ajustados  $\Delta r = r$ , como se hace en [187]. Siguiendo a autores como Becker, L., y Seshadri [18], la experimentación se apoyara en el uso de un rendimiento compuesto continuo. La principal ventaja de utilizar este enfoque, en este contexto particular, frente a una alternativa basada en rendimientos brutos, es el hecho de que es más fácil escalar el sistema. Esto es debido a que al aplicar esta simplificación se reduce la complejidad de cálculo de los rendimientos del mercado, pasando de  $n$  períodos que ejecutan  $O(n)$  operaciones de multiplicación, a una  $O(1)$  simple adición, es decir se pasa de un valor de complejidad lineal a otro de complejidad constante. Esto reduce el coste computacional de manera muy significativa, mientras que la aproximación del resultado obtenido sigue siendo muy buena.

Para obtener la aptitud de una regla de inversión representada por el individuo, a esta se le aplican todos los datos diarios de sesión sucesivamente, acumulando los rendimientos de inversión del mercado en los períodos en los que la regla marca que se ha de estar dentro del mismo, o los rendimientos de inversión libres de riesgo procedentes del mercado monetario, cuando, por el contrario, la regla señala estar fuera del mercado. La forma simple de rendimiento de la inversión, si se realiza en una única transacción, es la dada por el conjunto de ecuaciones 4-4, 4-5 y 4-6.

$$\pi_i = \frac{P_{si}}{P_{bi}} \times \left( \frac{1-c}{1+c} \right) - 1 \quad (4-4)$$

$$= \exp \left[ \sum_{t=b_i+1}^{s_i} r_t \right] \times \left( \frac{1-c}{1+c} \right) - 1 \quad (4-5)$$

$$= \exp \left[ \sum_{t=b_i+1}^{s_i} r_t + \ln \left( \frac{1-c}{1+c} \right) \right] - 1 \quad (4-6)$$

donde :

- $\pi_i$  es el rendimiento en una única transacción.
- $P_{si}$  es el precio de venta en la transacción.
- $P_{bi}$  es el precio de compra en la transacción.
- $c$  es el coste de una transacción expresado como una fracción del precio de compra o venta.

Mientras que el rendimiento compuesto continuo de una regla de inversión viene dado por la expresión 4-7.

$$r = \sum_{t=1}^T r_t \cdot I_b(t) + \sum_{t=1}^T r_f(t) \cdot I_s(t) + n \cdot \ln \left( \frac{1-c}{1+c} \right) \quad (4-7)$$

donde:

- $r_t = \ln(P_t) - \ln(P_{t-1})$  es el valor del rendimiento compuesto continuo.  $P_t$  es el precio en el momento  $t$ .  $r_t$  calcula el retorno de la inversión a lo largo del tiempo cuando, según la norma obtenida, el inversor debe estar en el mercado.
- $I_b(t)$  indica una señal de compra. Adopta el valor "1" si la regla sugiere comprar en el momento  $t$  y "0" de lo contrario.
- $r_f$  calcula el retorno de la inversión sin riesgo cuando, siguiendo la regla obtenida, el inversionista está fuera del mercado.
- $I_s(t)$  Indica una señal de venta. Es el valor opuesto a la señal  $I_b(t)$ . Adopta el valor "1" si la regla recomienda vender en el momento  $t$  y "0" en caso contrario.
- El tercer componente en la ecuación modela los costes de transacción (se asume que son del 0.25%),  $n$  denota el número de transacciones expresadas como una señal de compra seguida de una señal de venta (cualquier posición abierta se cierra el último día), y  $c$  es el coste de una transacción expresada como una fracción del precio.

$r_f$  se puede calcular de diferentes maneras. Jansen [102] lo calcula siguiendo la fórmula 4-8.

$$r_{f(t)} = \ln \frac{(1 + r_{f, \text{monthly}})}{\delta} \quad (4-8)$$

Donde:

- $(1 + r_{f, \text{mensual}})$  se refiere al interés mensual en el mercado monetario.
- $\delta$  a la cantidad de días en que el mercado está abierto.

En este trabajo se sigue el mismo enfoque, pero se considera, para simplificar, los días naturales en vez de los días laborables para el mercado. Es poco probable que esta decisión tenga algún impacto relevante, especialmente en un entorno de baja tasa de interés. Como Jansen [102] menciona en su tesis, “el cálculo del rendimiento no es absolutamente muy preciso, pero en cualquier caso los rendimientos obtenidos por las reglas de la PG en este concepto son marginales. Incluso algunos autores los descartan”.

Otros autores como Allen y Karjalainen [8] o Lohpetch y Corne [120], evalúan las soluciones en términos de exceso de inversión sobre la estrategia B&H  $\Delta r = r - R_{bh}$ . Aquí  $R_{bh}$  (RBH) constituye el retorno de la inversión obtenida comprando al inicio del período considerado y vendiendo al final del mismo, que se define formalmente en 4-9 como:

$$R_{bh} = \sum r_t + \ln\left(\frac{1-c}{1+c}\right) \quad (4-9)$$

En donde, como se discutió con respecto a la ecuación (4-7),  $r_t$  es el rendimiento del mercado durante el período  $t$ , y  $c$  representa el coste de transacción en una sola operación.

El rendimiento simple se puede calcular de nuevo, a partir del rendimiento compuesto continuo siguiendo la fórmula 4-10:

$$\pi_i = e^r - 1 \quad (4-10)$$

El enfoque seguido en esta tesis, al igual que en [187], es muy parecido, la única diferencia es que en este trabajo no se tiene en cuenta la resta de una constante,  $R_{bh}$ .

La representación de la función de aptitud, como se sugiere en esta tesis, tiene ventajas prácticas, ya que tiene un costo computacional ligeramente menor y una menor probabilidad de tener que lidiar con aptitudes negativas.

En la figura 4.1 se aporta un diagrama de flujo de cómo debería calcularse el valor de aptitud en la función de evaluación de los individuos, en el algoritmo de CE utilizado.

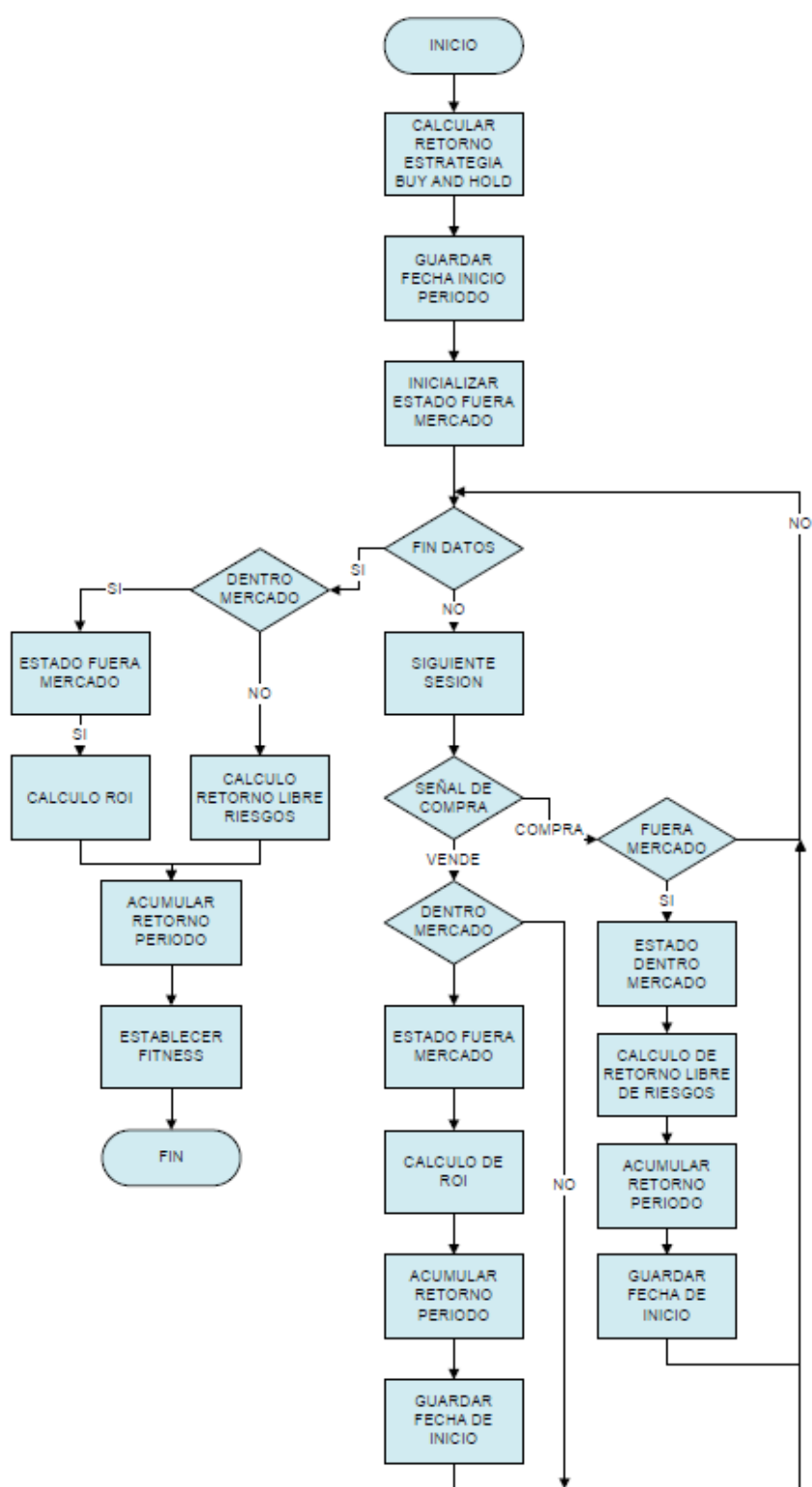


Figura 4.1: Función de aptitud (Fitness).

#### 4.4 Parametrización

Hay muchas posibilidades de parametrización para los algoritmos de PG y GE. En este estudio se han adoptado, en general, los valores más comunes encontrados en la literatura, y en otros casos, o en paralelo la elección de estos, se han elegido a partir de pruebas exploratorias realizadas en experimentos previos. En particular, se han tomado como referencia algunos de los valores usados por Koza [112] y que se presentan en la tabla 4.4.

**Tabla 4.4:** *Parametrización de la PG usada por Koza.*

Parámetros	Valor
Profundidad inicial máxima	6
Máxima profundidad	17
Generaciones	50
Tamaño de la población	500
Tamaño del torneo	2
Probabilidad de reproducción	0,1
Probabilidad de cruce	0,9
Probabilidad de mutación	0,01
Inicialización inicial	Ramped half-and-half
Algoritmo de selección	Proporcional al valor de aptitud
Probabilidad selección nodos terminales	20 %

Los experimentos de referencia, para las configuraciones de PG y las basadas en GE, utilizan tamaños de poblaciones que, dependiendo del experimento, se mueven en el rango de 500 a 3.000 individuos. La población inicial así definida, se ha evolucionado a lo largo de 50 ciclos evolutivos, siendo el número de generaciones transcurridas la condición de parada establecida, para la finalización del algoritmo y la obtención de resultados.

En la evolución se ha implementado la técnica exploratoria de elitismo, pero de un solo individuo. El procedimiento ha consistido en llevar la mejor estrategia obtenida de cada generación sin modificación a la siguiente, sustituyendo al

anterior élite cuando el individuo de la generación en curso mostraba una mejor aptitud. Se ha elegido un elitismo de un solo individuo, para preservar la carga genética de la mejor solución candidata, y al mismo tiempo evitar una convergencia prematura.

Otro elemento en común a la PG y las GE que debe tenerse en cuenta, y que persigue evitar una convergencia demasiado rápida, es un mecanismo diseñado para mejorar la variabilidad presente en la población, consistente en evitar individuos duplicados. Este mecanismo está dirigido, en primer lugar, a la fase de inicialización de la población de ambos algoritmos de PG y GE y, en segundo lugar, al operador genético de mutación. Si en el transcurso de cualquiera de estas dos operaciones un individuo aparece más de una vez en la población, porque ya estuviera presente en la misma, se reintentará la inicialización y sustitución del individuo repetido por uno nuevo, o se probará a eliminar el duplicado aplicando una nueva mutación forzada sobre el individuo original, según el caso. El número de intentos máximo, para generar un intercambio de una regla de inversión duplicada, está limitado a 100 intentos. Una vez alcanzado este límite, el individuo generado es simplemente aceptado.

En cuanto a los procesos computacionales que guían los algoritmos, algunos de ellos se han ejecutado en paralelo aprovechando las capacidades del hardware ofrecido. En este trabajo se han creado ocho hilos de ejecución (threads ligeros)<sup>3</sup> dedicados a la evaluación de la aptitud, y otros ocho se han empleado en los procesos de creación de las nuevas poblaciones. La gestión de la memoria también se ha paralelizado mediante la parametrización del recolector de basura de la máquina virtual, forzándole a usar ocho hilos de ejecución<sup>4</sup>.

---

<sup>3</sup> La razón de haber elegido ocho hilos de ejecución, es que las máquinas utilizadas solo contaban con cuatro núcleos de dos unidades funcionales cada uno. Dado que java crea un hilo de espacio de usuario, no está limitado al número de núcleos de la máquina, sino que se pueden crear tantos como se quieran pudiendo llegar a bloquear al sistema [76]. En resumen, se ha decidido establecer ocho teniendo en mente la disponibilidad física de los equipos de prueba.

<sup>4</sup> Para ello se ha usado el parámetro de JVM `XX:ParallelCMSThreads = 8`, que obliga al recolector de basura a usar 8 hilos. Se ha decidido usar el recolector de basura CMS, en lugar del nuevo G1, ya que en las pruebas se vio que aunque con G1 los tiempos de latencia en las paradas de recolección eran menores para G1, del orden de un 20 % aproximadamente, el número de paradas era tres veces superior, eliminando por completo su ventaja competitiva.

#### 4.4.1 Parametrización en la PG

El resto de parámetros específicos de la PG, utilizados en las experimentaciones, los resumimos en esta sección.

La inicialización de la primera población se realiza mediante el empleo del algoritmo *Uniform*, definiendo un valor deseado de complejidad inicial de entorno a cinco nodos y de un máximo de cincuenta, valor que también se empleará en la mutación.

En la operación genética de cruce el valor máximo de complejidad utilizado es igualmente de cincuenta. No existe límite inferior, pero si una profundidad máxima, que, como en la mutación, es de siete. En cuanto a la probabilidad de selección de nodos terminales se ha optado por un valor de un 20 %, y para la de los no terminales de un 80 %, en consonancia con los valores típicos encontrados en literatura.

Por otro lado, los valores de profundidad y de complejidad máxima del árbol del individuo entran en competición entre sí, y en la evolución del algoritmo estos límites se suelen alcanzar rápidamente. Dependiendo del tipo de problema uno predominará sobre otro. Los valores elegidos en este trabajo están basados en la hipótesis de que la solución se encuentra en niveles de baja complejidad y profundidad. Por lo que la sospecha de que los árboles con un cierto tamaño proporcionarán una mayor probabilidad de alcanzar los mejores resultados, se ha utilizado en esta tesis para guiar al algoritmo en la evolución.

En cuanto al sistema de selección elegido para la reproducción y cruce, basado en un método ordinal, es el de torneo simple con dos candidatos, excepto en las experimentaciones en las que se haga uso de parsimonia, en cuyo caso el torneo es de tipo parsimonia lexicográfica de tamaño dos.

Para la operación de reproducción se ha utilizado una tubería de reproducción de PG múltiple con las siguientes tres fuentes de entrada y sus probabilidades de uso relacionadas<sup>5</sup>:

- **Cruce.** Con una probabilidad de un 0,8 se seleccionan de la población, por torneo, dos individuos padres. Estos se cruzan a continuación entre sí, gene-

---

<sup>5</sup> El uso de las tuberías comporta una parametrización adicional. Los valores de probabilidad utilizados son los sugeridos por Sean Luke [129].



rando otros dos individuos hijos que se introducen en la nueva población. En el operador genético de cruce se establece un límite deseado de complejidad del árbol de los hijos de cincuenta nodos y una profundidad máxima de siete niveles.

El cruce que se realiza es de tipo subárbol estándar: esto es, por cada uno de los individuos seleccionados se elige un nodo de sus árboles, y se intercambian entre sí los dos subárboles fijados a esos dos nodos padre.

- **Reproducción por copia.** Con una probabilidad de un 0,1, un individuo que ha sido seleccionado por torneo se copia directamente, sin mutación, en la nueva población.
- **Mutación.** Con una probabilidad de un 0,1, un individuo que ha sido obtenido por torneo es mutado e introducido en la nueva población. El operador genético de mutación es del tipo estándar, que realiza una mutación de subárbol simple. El proceso es como sigue: se selecciona un nodo del árbol del individuo elegido, y el subárbol escogido del que cuelga ese nodo se reemplaza en su totalidad por un nuevo árbol generado aleatoriamente. El número mínimo de nodos deseados, en el individuo resultante, se establece en diez y el máximo en cincuenta.

Los nuevos individuos se agregan a la población mediante una estrategia de reemplazo  $(\mu, \lambda)$  [175], manteniendo el tamaño de la población constante a través de las generaciones. El mejor individuo de la generación en curso es introducido en la nueva población sin sufrir ningún cambio, siguiendo una estrategia de tipo elitista.

Por último, la variante de la técnica de la PG utilizada es la STGP, por lo que se definen los árboles de restricciones de tipos, que definen los tipos de datos de los parámetros y retornos de cada nodo del árbol, especificando: el tipo de retorno del nodo raíz, el tipo de retorno de los nodos intermedios y tipos esperados de sus nodos hijos, así como el tipo de retorno de los nodos hojas finales.

#### 4.4.2 Parametrización de las GE

Las GE, básicamente descansan en la misma parametrización discutida para la PG. No obstante, existen algunas diferencias con ella, que se pasa a exponer a

continuación:

- La inicialización de los individuos de la población se define a partir de series geométricas, en las que se establece un valor de complejidad inicial mínimo de cinco y una probabilidad de crecimiento de 0,85.
- Por defecto, en la formación de los individuos se permiten dieciséis duplicaciones circulares, de su vector de enteros, a partir de la gramática que define su posible estructura. Esto se implementa de la siguiente forma: si se llega al final de la lista de enteros generada aleatoriamente, sin que el individuo haya terminado de formarse según la gramática, el proceso vuelve a utilizar la lista otra vez desde el principio, continuando de forma circular. Si tras esos dieciséis intentos no ha sido posible construir el individuo, este muere asignándole el menor valor de aptitud posible.
- El valor que pueden adoptar los genes, dentro del vector de enteros que define el genotipo del individuo, está incluido en el margen que va de 0 a 79. (80 valores en total). Valor definido como el máximo común múltiplo de todas las producciones de las reglas gramaticales, que forma parte de la gramática asociada al problema.

En las GE también se ha definido una tubería de reproducción múltiple, como en la PG, con las siguientes tres fuentes de entrada y sus probabilidades de uso relacionadas:

- **Cruce.** Con una probabilidad de un 0,85, se seleccionan dos individuos por torneo y se cruzan mediante un único punto de cruce, generando, a partir de ellos, otros dos individuos que se introducen en la nueva población.

El procedimiento de cruce de vectores selecciona dos índices aleatorios  $i$  y  $j$  en cada uno de los dos individuos padre A y B, y luego se intercambia la cadena de genes desde el punto de inicio del índice hasta el final del vector,  $A_i, \dots, A_{final}$  y  $B_j, \dots, B_{final}$ .

Finalmente, a los individuos obtenidos se les aplica un mecanismo de reparación por truncamiento, que empieza por determinar cuántos enteros de su genotipo vectorial se consumen, partiendo de la definición gramatical, en la generación del árbol, y termina eliminando el resto de los enteros del vector que no se han utilizado.

- **Duplicación.** Con una probabilidad de un 0,05, se aplica un mecanismo de duplicación a un solo individuo seleccionado por el sistema de torneo. En la duplicación se selecciona una secuencia de enteros del propio vector del individuo, definida por dos índices aleatorios, y se anexa al final del vector. Luego se aplica la técnica de truncamiento.
- **Mutación.** Con una probabilidad de un 0,1, un individuo es seleccionado por el método de torneo y se muta con una probabilidad de 0,05. La mutación se realiza modificando aleatoriamente uno de sus genes, asignándole un valor comprendido entre los valores máximos y mínimos definidos.

Se utiliza una mutación de tipo uniforme permitiendo una duplicación circular del vector de genes, de hasta dieciséis intentos, en el proceso de traducción gramatical del individuo mutado. El resto de los individuos, con una probabilidad de un 0,95, se copian sin modificación. Finalmente, los individuos mutados, previamente truncados, se introducen en la población.

Al igual que en la PG, el método de selección elegido para la reproducción y cruce es el de torneo simple con dos candidatos, excepto en las experimentaciones en las que se haga uso de parsimonia en cuyo caso el torneo es de tipo parsimonia lexicográfica de tamaño dos.

#### 4.4.3 Generación de aleatoriedad

Un aspecto crítico de un algoritmo genético es que funciona como un sistema estocástico. Por ello, necesitamos generar de forma aleatoria una población inicial de individuos, con la mayor diversidad genética posible, en la esperanza de cubrir al máximo y de la mejor forma posible el espacio de soluciones<sup>6</sup>. En apoyo de esta inicialización, disponemos en la mayoría de los lenguajes de una función generadora de números aleatorios<sup>7</sup>. Prestar atención a la elección de un generador aleatorio, que sea de alta calidad, así como cuidar de su modo de empleo correcto, es un aspecto vital en la implementación de las GE o de la PG.

En esta tesis, se hace uso, dentro del marco de trabajo de ECJ, del generador de números pseudoaleatorios de Mersenne Twister (Mersenne Twister pseudorandom

<sup>6</sup> En los procesos de mutación o selección es también muy importante.

<sup>7</sup> Realmente se trata de generadores de números pseudoaleatorios.

number generator PRNG MT) ideado por Makoto Matsumoto y Takuji Nishimura,<sup>89</sup> [136], con una semilla de trabajo diferente para cada uno de los hilos de ejecución empleados, y que, para evitar que sea una simple secuencia del reloj utilizado, son inicializados en cada uno de los experimentos, aun cuando estos sean ejecutados en procesos por lotes.

---

<sup>8</sup> Su nombre deriva de la longitud de su período, que se corresponde con un número primo de Mersenne. Existen diferentes implementaciones, distinguiéndose entre ellas por el tamaño de los números primos de Mersenne utilizados. El más popular es el Mersenne Twister MT19937.

<sup>9</sup> Un número primo de Mersenne es un número primo entero positivo  $m$  cuyo valor es una unidad menor que una potencia entera positiva de 2,  $M_n = 2^n - 1$ .

## Capítulo 5

# Metodología de evaluación

En este apartado, se tratarán los aspectos relacionados con la elección del tamaño de la muestra utilizada en los experimentos, así como del protocolo de evaluación de significación estadística aplicado a las experimentaciones realizadas, en las tres partes de las que se compone esta tesis.

### 5.1 Elección del tamaño de la muestra

En la primera parte del trabajo se ha realizado una comparación entre las técnicas de las GE y de la PG, para contrastar su comportamiento y rendimiento en la búsqueda de reglas de inversión efectivas que batiesen a la estrategia B&H. Se pretendía que, en este experimento, el error cometido fuese muy bajo y con un alto nivel de significación estadística. En el tratamiento estadístico de la experimentación la población se puede considerar como de tamaño infinito, pues se pueden repetir los experimentos tantas veces como se quiera. Precisamente por esta razón, de no poder observar toda la población, cometeremos siempre un error de muestreo que, aunque es inevitable al existir siempre una diferencia entre los valores medios desconocidos de esa población, podemos reducir. El minimizarlo depende directamente del tamaño de la muestra, e inversamente de la dispersión o desviación estándar del estimador<sup>1</sup>.

El error máximo permitido en el cálculo de un intervalo de confianza, para la media, viene dado por la fórmula 5-1 [201].

---

<sup>1</sup> Un estimador es un estadístico, o función de la muestra, usado para estimar un parámetro desconocido.

$$E = \frac{Z_{\alpha/2} \cdot \sigma}{\bar{x} \cdot \sqrt{n}} \quad (5-1)$$

Donde:

$E$  es el error máximo permitido.

$Z_{\alpha/2}$  es el estadístico para el nivel de confianza deseado, en una distribución normal tipificada.

$\sigma$  se corresponde con la desviación estándar.

$n$  es el tamaño de la muestra.

$\bar{x}$  es la media muestral.

Para disminuir el error, una vez escogido el nivel de significación, basta con despejar  $n$  de la ecuación 5-1.

$$n = \left( \frac{Z_{\alpha/2} \cdot \sigma}{\bar{x} \cdot E} \right)^2 \quad (5-2)$$

Donde el error máximo permitido, lo expresaremos como un valor  $\epsilon$  máximo, en porcentaje de la media calculada sobre la muestra obtenida, como se expone en la ecuación 5-3.

$$n = \left( \frac{Z_{\alpha/2} \cdot \sigma}{\bar{x} \cdot \epsilon} \right)^2 \quad (5-3)$$

Tras varias pruebas iniciales, para determinar el número de simulaciones necesarias, y en base al error obtenido en ellas, se ha fijado que se han de realizar al menos 2.000 experimentos para cada una de las técnicas de CE examinadas, con el

objetivo de intentar alcanzar un nivel de significación  $\alpha$  del 95 % y un error máximo  $\epsilon$  del 5 %.

En los otros dos conjuntos de experimentaciones realizados, relativos a los planteamientos sobre *las soluciones basadas en conjuntos de modelos en inversión algorítmica* y a *la generación dinámica de reglas de inversión*, el número de experimentos ejecutados se ha establecido en 30<sup>2</sup>.

## 5.2 Protocolo de evaluación de significación estadística

### 5.2.1 Intervalo de confianza

En el primer experimento, se han calculado los niveles de confianza de las medias de población de las muestras obtenidas con un nivel de confianza del 97.5 %. También se han realizado los contrastes de hipótesis mediante un procedimiento de verificación que toma como referencia la técnica central de GE, a la que se compara con el resto de técnicas de las GE y la PG. La probabilidad de aceptación de  $H_0 : \mu = \mu_0$ , en la estimación, se representa por  $(1 - \alpha)$  y se denomina nivel de confianza. En estas circunstancias,  $\alpha$  es el llamado error aleatorio o nivel de significación, y establece el valor de posibilidad de fallar en la estimación.

En el primer caso los niveles de confianza de las medias de las poblaciones se han obtenido aplicando las ecuaciones 5-4, 5-5, 5-6, 5-7, 5-8.

---

<sup>2</sup> En estos experimentos se esperaba una computación más extensa, que debía conciliarse con la necesidad de tamaños de muestra mínimos, que proporcionasen un poder adecuado para distinguir una distribución normal de otras distribuciones, de modo que el no poder encontrar una desviación significativa de la normalidad, nos lleve a un grado razonable de confianza de que los datos son consistentes con esa distribución. La falta de normalidad en la distribución de las poblaciones de las que derivamos las muestras no suele plantear un problema. La razón es que la distribución de diferencias entre dos medias muestrales independientes es a menudo lo suficientemente normal como para que las estadísticas sigan siendo válidas. La justificación se encuentra en el teorema del límite central [81], una "ley estadística de la gravedad", que afirma (en su forma más simple) que la distribución de la media de una muestra será aproximadamente normal siempre que el tamaño de la muestra sea lo suficientemente grande, las variables que intervienen sean independientes y la varianza sea finita pero no "0". El número de observaciones necesarias depende de la distribución de los valores de los datos en la población de donde proviene la muestra. Cuanto menos normal sea (lo que significa, por lo general, que sea más sesgado), mayor será el requisito de tamaño de la muestra. En la mayoría de los libros de textos se recomienda un valor de 30. En el caso que nos ocupa el número 30 también proviene de un examen de la distribución de chi-cuadrado. Para los datos distribuidos normalmente, y según el teorema central del límite deberían de serlo, se necesitan aproximadamente 30 observaciones (diferencias en las variables) para tener límites de confianza razonablemente cortos en la estimación de la varianza.

Cálculo de la media:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \quad (5-4)$$

Cálculo de la Varianza:

$$\sigma^2 = \sum_{i=1}^n \frac{(x_i - \bar{x})^2}{n - 1} \quad (5-5)$$

Intervalos de confianza:

$$C_1 = \bar{x} - z_{\frac{\alpha}{2}} \frac{\sigma}{\sqrt{n}} \quad (5-6)$$

$$C_2 = \bar{x} + z_{\frac{\alpha}{2}} \frac{\sigma}{\sqrt{n}} \quad (5-7)$$

A partir de ahí podemos calcular el intervalo de confianza donde se encontrará la media de la población  $\mu_{\bar{X}}$  con una significación de  $1 - \alpha$ . Donde la hipótesis de que la media de la muestra coincida con la de la población viene dada por  $H_0 : \mu = \mu_{\bar{X}}$ . Si el valor de  $\mu$  no está incluido en el intervalo 5-8 se tendrá que rechazar  $H_0$ .

$$P\left(\bar{x} - z_{\frac{\alpha}{2}} \frac{\sigma}{\sqrt{n}} \leq \mu \leq \bar{x} + z_{\frac{\alpha}{2}} \frac{\sigma}{\sqrt{n}}\right) = 1 - \alpha \quad (5-8)$$

Para el cálculo de contrastes de medias bilaterales, entre las diferentes medias de los experimentos realizados, se han aplicado las fórmulas 5-9,5-10, 5-11, 5-12.

Diferencia de las medias muestrales:

$$\bar{x} = \bar{x}_1 - \bar{x}_2 \quad (5-9)$$



Desviación típica:

$$\sigma_x = \sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}} \quad (5-10)$$

Intervalos de confianza:

$$C_1 = \bar{x} - Z_{\frac{\alpha}{2}} \sigma_x \quad (5-11)$$

$$C_2 = \bar{x} + Z_{\frac{\alpha}{2}} \sigma_x \quad (5-12)$$

Donde la hipótesis del contraste de medias se establece como  $H_0 : \mu = \mu_{\bar{x}}$ . Si el valor de  $\mu$  no está incluido en el intervalo dado por 5-11 y 5-12 se tendrá que rechazar la hipótesis nula  $H_0$ .

Aunque se ha realizado este contraste, entre los experimentos, y está disponible en el apéndice C, se ha seguido un protocolo de contraste de hipótesis más potente que se describe en la siguiente sección.

### 5.2.2 Contraste de hipótesis

La significación estadística de un experimento, parte de una hipótesis que es una asunción relativa a una característica de una o varias poblaciones. La hipótesis puede ser cierta o no, y para determinarlo se contrasta por medio de la información obtenida de las muestras experimentales. Tanto si estas se aceptan, como si se rechazan, se puede cometer un error de uno de los siguientes tipos: que la hipótesis se acepte, siendo falsa, o que por el contrario siendo cierta, se rechace.<sup>3</sup>

Cuando, como en este trabajo, se comparan dos poblaciones a partir de una variable cuantitativa continua, las pruebas de contraste, con frecuencia, se basan en la suposición de que la muestra aleatoria obtenida pertenece a una distribución de

---

<sup>3</sup> El primer error es conocido como error de tipo I o  $\alpha$  y el segundo como error de tipo II o  $\beta$ . Los tipos de errores  $\beta$ , por lo general, son función del tamaño muestral, y la única forma de minimizarlos, por tanto, es ampliando el tamaño de la muestra  $n$ . Los riesgos de los errores  $\alpha$  y  $\beta$  están inversamente relacionados y se determinan según el nivel de significancia y la potencia de la prueba, sólo pueden disminuirse los dos a la vez, aumentando  $n$ .

probabilidad de tipo normal. Sin embargo, esta suposición no es siempre válida o es difícil de comprobar cuando la muestra es muy pequeña.

En estos casos para el análisis de las muestras podemos utilizar pruebas estadísticas no paramétricas, que no se basan en ninguna suposición en cuanto a la distribución de probabilidad de la muestra aleatoria obtenida de la población, o bien podemos partir del análisis de la distribución de probabilidad, y si este análisis determina que hay normalidad aplicaremos pruebas de tipo paramétrico [195]. Por tanto, para poder realizar un protocolo de evaluación de significación estadística, lo habitual es seguir los siguientes pasos [161]:

1. Determinar la hipótesis nula  $H_0$  y la alternativa  $H_a$ <sup>4</sup>.
2. Elegir el nivel de significación deseado para el contraste. Se trata de fijar el valor  $\alpha$ , o probabilidad de cometer el error de tipo I en el contraste de hipótesis.
3. Especificación del tamaño muestral  $n$ . La muestra será del tipo  $(x_1, x_2, x_3 \dots x_n)$ .
4. Seleccionar un estadístico, o función de decisión, cuya distribución muestral sea conocida en el caso de que la hipótesis nula sea cierta. Habitualmente el estadístico  $T$  será una función del estimador natural asociada al parámetro que interviene en la hipótesis.
5. Determinar la región crítica. Conocida la distribución del estadístico bajo la hipótesis nula, se halla su valor crítico en el punto  $\alpha$  es decir hallar un  $k/P_0(T > k = \alpha)$ . La región crítica será  $T(x_1, x_2, x_3 \dots x_n) > k$ .<sup>5</sup>
6. Calcular el valor del estadístico de contraste para la muestra elegida.  $T_0 = T(x_1, x_2, x_3 \dots x_n)$ .
7. Sacar las conclusiones estadísticas del contraste (aceptar o rechazar  $H_0$ ). Se rechaza  $H_0$  si el valor de la función de decisión cae dentro de la región crítica ( $T_0 > k$ ), o bien aceptándola en caso contrario ( $T_0 \leq k$ ).

---

<sup>4</sup> La hipótesis nula,  $H_0$ , es una afirmación que no se rechaza, a menos que los datos de la muestra parezcan demostrar que es falsa. Si los resultados de la muestra no apoyan la hipótesis nula, se rechaza la hipótesis y la conclusión aceptada se llama hipótesis alternativa  $H_a$  o  $H_1$ . La hipótesis nula es la que por tanto se acepta o rechaza como conclusión del contraste.

<sup>5</sup> De esta manera hay una mayor firmeza cuando se rechaza una hipótesis, razón por la cual se fija en  $H_0$  lo que se quiere negar. Cuando no se refuta  $H_0$  no se ha demostrado el contrario, simplemente no se ha podido rechazar.

## 8. Interpretación práctica de los resultados.

Los resultados obtenidos han sido evaluados siguiendo el protocolo utilizado por García et al. [74] que se detalla a continuación en el algoritmo 5-1.

El contraste de la normalidad, en la distribución de probabilidad de la población formada por los retornos de inversión (variables aleatorias continuas), se realiza mediante la prueba de *Kolmogorov-Smirnov*, aplicando la corrección de *Lilliefors*. Si se rechaza la normalidad de las observaciones, se aplica la prueba no paramétrica de los rangos de signos de *Wilcoxon*. De lo contrario, la prueba de *Levene* controla la homocedasticidad<sup>6</sup> de las variaciones. Si hay *homocedasticidad*, se aplica la prueba *t-student*, y si hay heterocedasticidad, se usa la prueba *t* de *Welch*.

La región crítica, que permite rechazar la hipótesis nula, se establece mediante contrastes unilaterales (una cola):

$$H_1 : \theta > \theta_1 \text{ y } H_1 : \theta < \theta_1 \quad (5-13)$$

---

**Algoritmo 1:** Protocolo de evaluación de significación estadística.
 

---

```

if los retornos siguen una distribución normal (test de Kolmogorov-Smirnov con la
corrección de Lilliefors). then
    if la varianza es homogénea (test de Levene). then
        Se aplica el test de t-student.
    else
        Se ejecuta el test de Welch.
    end if
else
    Se aplica el test de Wilcoxon para comparar medias.
end if
  
```

---

<sup>6</sup>Se habla de homocedasticidad si el error cometido por el modelo predictivo tiene siempre la misma varianza.

Las pruebas de *Kolmogorov-Smirnov* [109], son pruebas no paramétricas que no se basan en suposiciones sobre la existencia de parámetros de distribución, como la media y la desviación estándar de la población subyacente. Eso significa que si tenemos datos extraños que no parecen encajar en ninguna distribución identificable, o bien definida, todavía podemos hacer pruebas estadísticas.<sup>7</sup>

El estadístico de *Kolmogorov-Smirnov* para una función de distribución acumulativa dada  $S_n(x)$  viene dado por la fórmula:

$$D_n = \max_x |F(x) - S_n(x)| \quad (5-14)$$

Donde:

$S_n(x)$  función de distribución acumulativa de la población muestral.<sup>8</sup>

$F(x)$  función de distribución acumulativa de la población muestral, una vez normalizada, suponiendo que la muestra original sigue una distribución normal.

$\max_x$  es la máxima distancia de las diferencias.

La hipótesis nula  $H_0$  establece que la muestra se extrae de la distribución de referencia. Si los datos se distribuyen normalmente el valor crítico  $D_{n,\alpha}$ , tomado de las tablas *Kolmogorov-Smirnov*, será mayor que  $D_n$ .

En estadística, la prueba de *Lilliefors* [117] es una prueba de normalidad basada en la prueba de *Kolmogorov-Smirnov* <sup>9</sup> [6]. Sin embargo, cuando la media de la población y la desviación estándar no se conocen, sino que se estiman a partir de los datos de la muestra, la prueba de *Kolmogorov-Smirnov* basada en los valores críticos de las tablas de *Kolmogorov-Smirnov* arrojan resultados que son demasiado conservadores.

<sup>7</sup> Una alternativa a la prueba de *Kolmogorov-Smirnov* con la corrección de *Lillieford* es la prueba de *Anderson-Darling*. Se trata de una prueba no paramétrica que determina si los datos de una muestra provienen de una distribución específica [11].

<sup>8</sup> La función de distribución acumulativa de una variable aleatoria de valor real  $X$ , o simplemente la función de distribución de  $X$ , evaluada en  $x$ , es la probabilidad de que  $X$  tome un valor menor o igual que  $x$ .

<sup>9</sup> Cuando la muestra es como máximo de un tamaño de 50 se puede contrastar la normalidad con la prueba de *shapiro* o *Shapiro-Wilk* [188].

*Lilliefors* creó una prueba relacionada que brinda resultados más precisos. La prueba de *Lilliefors* usa los mismos cálculos que la prueba de *Kolmogorov-Smirnov*, pero se usa la tabla de valores críticos de *Lilliefors* en lugar de la tabla de *Kolmogorov-Smirnov*. Como los valores críticos en esta tabla son más pequeños, es menos probable que la prueba de *Lilliefors* muestre que los datos de la muestra se distribuyen normalmente.

En el caso de que los datos no provengan de una distribución normal, o cuando estos tienen un tamaño demasiado reducido para poder determinar si realmente proceden de poblaciones normales, se suele utilizar la prueba de los rangos con signo de *Wilcoxon* [214]. Esta prueba no paramétrica compara el rango medio de dos muestras relacionadas, y determinar si existen diferencias entre ellas. El test compara la mediana de las diferencias bajo determinadas condiciones. En general, la prueba de rangos con signo de *Wilcoxon* (*Wilcoxon signed-rank test*) compara si las diferencias entre pares de datos siguen una distribución simétrica entorno a un valor. Si dos muestras proceden de la misma población es de esperar que las diferencias entre cada par de observaciones se distribuyan de forma simétrica entorno al cero. Suele considerarse el equivalente no paramétrico del *t-test*, aunque trabaja con medianas en vez de medias. En el protocolo definido se aplica cuando se cumple que:

1. Las muestras experimentales obtenidas no siguen una distribución normal.

El método supone que:

1. Los datos pareados de las muestras proceden de la misma población.
2. Las parejas son elegidas de forma aleatoria e independiente.
3. Los datos se miden dentro de un intervalo.

Procedimiento: Determinar si para  $n$  pares de observaciones, de dos muestras de  $n$  observaciones, para los pares ( $i = 1, \dots, n$ ) cuyas medidas son  $(x_{1,i}, x_{2,i})$ , se puede decidir que los valores  $x_{1,i}$  e  $x_{2,i}$  son o no iguales.

$H_0$ : Hipótesis nula. La diferencia entre los pares sigue una distribución simétrica alrededor de cero.

$H_1$ : Hipótesis alternativa. La diferencia entre los pares no sigue una distribución simétrica alrededor de cero.

1. Para  $(i = 1, \dots, n)$ , calcular  $|x_{2,i} - x_{1,i}|$  y el  $\text{signo}(x_{2,i} - x_{1,i})$ , donde  $\text{signo}$  es la función de signo.
2. Excluir los pares  $|x_{2,i} - x_{1,i}| = 0$ .  $n_r$  es el tamaño reducido de la muestra resultante.
3. Ordenar los restantes  $N_r$  de menor a mayor, en base a  $|x_{2,i} - x_{1,i}|$ .
4. Clasifíquense los pares, comenzando asignando al más pequeño el valor "1". Los empates reciben un rango igual al promedio de los rangos que abarcan.  $R_i$  denota el rango resultante.
5. Calcúlese el estadístico de suma de los rangos con signo:

$$w = \sum_{i=1}^{n_r} [\text{signo}(x_{2,i} - x_{1,i}) \cdot R_i] \quad (5-15)$$

6. Bajo hipótesis nula,  $w$  sigue una distribución específica con un valor esperado de "0" y una varianza:

$$(\sigma_w)^2 = \frac{n - r(n_r + 1)(2n_r + 1)}{6} \quad (5-16)$$

7.  $w$  se compara con un valor crítico de una tabla de referencia para  $w$ . La prueba, de una o dos colas, consiste en rechazar  $H_0$  si  $|w| > w_{\text{critico}, N_r}$ .
8. Para  $N_r < 20$  se aplica la prueba original usando la estadística  $T$  [194], que se obtiene de la menor de las dos sumas de rangos de un signo dado en valor absoluto.<sup>10</sup>

<sup>10</sup> La tabla de referencia de *Wilcoxon* "Critical Values for the Wilcoxon Signed Rank Test Statistics T" se puede obtener de <https://www.safaribooksonline.com/library/view/nonparametric-statistics-a/9781118840429/bapp02.xhtml?orpg>, última visita 29/01/2019.

9. Si  $n_r$  aumenta considerablemente, la distribución de  $w$  converge hacia una distribución normal (valores de  $n_r > 20$ ), lo que permite calcular el rango de  $z$  como:

$$z = \frac{W}{\sigma_w} \quad (5-17)$$

y donde

$$\sigma_w = \sqrt{\frac{n - r(n_r + 1)(2n_r + 1)}{6}} \quad (5-18)$$

Para realizar una prueba de una cola, se rechaza  $H_0$  si  $|z| > z_{critico}$ . Alternativamente, la prueba de una sola cola se puede realizar con la distribución exacta o aproximada. Los valores de la probabilidad observada  $p$  también se pueden calcular y utilizarse.

La prueba inferencial de *Levene* [98], evalúa el supuesto de que las varianzas de las poblaciones, de las que se extraen diferentes muestras, son iguales (homocedasticidad). Si *Levene* concluye, que las diferencias obtenidas en las variaciones de la muestra no se han producido, sobre la base de un muestreo aleatorio de una población con varianzas iguales, la hipótesis nula de igualdad de varianzas se rechaza, y se concluye que hay una diferencia entre las variaciones en la población. Su estadístico se calcula de la siguiente forma:

$$w = \frac{(n - k) \sum_{i=1}^k n_i (\bar{Z}_i - \bar{Z}_{..})^2}{(k - 1) \sum_{i=1}^k \sum_{j=1}^n (Z_{ij} - \bar{Z}_i)^2} \quad (5-19)$$

En donde:

$w$  es el resultado obtenido en la prueba.

$k$  es el número total de diferentes grupos a los que pertenecen los casos muestreados.

$n$  es el número total de casos obtenidos en todos los grupos. Para el caso de dos grupos  $n$  es simplemente  $(n_1 + n_2)$ .

$n_i$  es el número de casos en el grupo  $i$ .

$\bar{Z}_i$  es la media aritmética de los valores del grupo  $i$ .

$\bar{Z}_.$  es la media aritmética de los valores de todos los grupos.

$Y_{ij}$  es el valor de la variable medida para el  $j$  esimo caso del  $i$  esimo grupo.

$\bar{Y}_i$  es la media aritmética del  $i$  esimo grupo.

Y el número de grados de libertad<sup>11</sup> utilizados para la prueba de significancia se establece como:

$$g = n_1 + n_2 - 2 \quad (5-20)$$

La prueba  $t$  de *Welch* [211], o la prueba  $t$  de varianzas desiguales, es una prueba aplicada a dos muestras que se usa para probar la hipótesis de que dos poblaciones tienen medias iguales. La prueba  $t$  de *Welch* es una adaptación de la prueba  $t$  de *Student*, es más fiable cuando las dos muestras tienen varianzas y tamaños de muestra desiguales. Estas pruebas a menudo se denominan pruebas  $t$  “no pareadas” o “muestras independientes”, ya que generalmente se aplican cuando las unidades estadísticas subyacentes a las dos muestras que se comparan no se superponen. En el protocolo definido se aplica cuando se cumple que:

1. Las muestras experimentales obtenidas tiene una distribución normal.
2. Las varianzas de ambas muestras no son homogéneas. Heterocedasticidad.

<sup>11</sup> Los grados de libertad vienen determinados por el número de valores que pueden ser asignados de forma arbitraria (conjunto de observaciones), antes de que el resto de las variables adopten el valor, a partir de estas. Se calculan mediante la fórmula  $n - r$ . Donde  $n$  es el número de observaciones de la muestra que pueden adoptar un valor y  $r$  es el número de variables cuyo valor dependerá del que tomen los miembros de la muestra que son libres.



Dado que en este trabajo las muestras tienen el mismo tamaño  $n = n_1 = n_2$ , el estadístico  $t$  se puede calcular como:

$$t = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}}} \quad (5-21)$$

$$t = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{\sigma_1^2 + \sigma_2^2}{n}}} \quad (5-22)$$

Donde:

$\bar{X}_1 - \bar{X}_2$ , constituye la diferencia de medias de ambas muestras.

$n$ , es el tamaño de la muestra.

$\sigma_1, \sigma_2$ , son las varianzas de ambas muestras.

Y el número de grados de libertad utilizados para la prueba de significancia se calcula por:

$$g = \frac{\left(\frac{\sigma_1^2 + \sigma_2^2}{n}\right)^2}{\frac{\sigma_1^4 + \sigma_2^4}{n^2(n-1)}} \quad (5-23)$$

La distribución  $t$  de *Student* [160], estima la media de una población distribuida normalmente, en situaciones donde el tamaño de la muestra es pequeño y la desviación estándar de la población es desconocida. En el protocolo definido se aplica cuando se cumple que:

1. Las muestras experimentales obtenidas tiene una distribución normal.
2. Las varianzas de ambas muestras son homogéneas. Homocedasticidad.

De nuevo si consideramos que las muestras tienen el mismo tamaño  $n = n_1 = n_2$  el estadístico  $t$  se puede calcular como:

$$t = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}}} \quad (5-24)$$

$$t = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{\sigma_1^2 + \sigma_2^2}{n}}} \quad (5-25)$$

Donde:

$\bar{X}_1 - \bar{X}_2$ , constituye la diferencia de medias de ambas muestras.

$n$ , es el tamaño de la muestra.

$\sigma_1, \sigma_2$ , son las varianzas de ambas muestras.

Y el número de grados de libertad utilizados para la prueba de significancia viene dado por:

$$g = n_1 + n_2 - 2 \quad (5-26)$$

Por último, sea cual sea el estadístico utilizado para efectuar los contrastes de hipótesis, es necesario determinar la región crítica asociada a un nivel de significación prefijado, o alternativamente, el  $p$ -valor asociado al valor del estadístico de prueba utilizado en dicho contraste. Ambos valores se buscarán en la distribución de probabilidad, que siga el estadístico de prueba de dicho contraste, bajo la hipótesis nula  $H_0$ .

En general se prefiere el uso de  $p$ -valor, porque es el valor de significación observado, o valor de probabilidad  $p$ , correspondiente al valor de significación estadística  $\alpha$  más pequeño que nos señale que la hipótesis  $H_0$  debe ser rechazada. Tiene la ventaja de que este valor indica el riesgo de cometer un error de tipo I y

en el estudio se pueden analizar las decisiones en función de la selección de  $\alpha$ . En el protocolo estadístico de este trabajo se hará uso de *p-valor*.

*Esta página se ha dejado en blanco intencionadamente.*

## Capítulo 6

# Selección de la técnica apropiada de computación evolutiva

### 6.1 Introducción

La finalidad de este primer planteamiento es realizar un trabajo de evaluación, en el que compararemos dos de los algoritmos más populares para la generación de reglas de inversión basados en CE. El objetivo principal que se persigue es seleccionar aquella técnica, que tenga un mejor comportamiento en la búsqueda de reglas de inversión, a partir del análisis de series temporales financieras. La elección es importante, pues a partir de ella se pretende extraer reglas de inversión que sean: rentables, eficientes, robustas y fáciles de interpretar, a través de soluciones basadas en conjuntos de modelos inversión y de sistemas de generación de modelos dinámicos.

Se persigue, en primer lugar, evaluar varias características fundamentales de las técnicas candidatas: la primera la eficiencia de los algoritmos básicos de las GE y de la PG en términos de retornos de inversión; la segunda el impacto de agregar, a los algoritmos, técnicas que hacen uso de funciones definidas automáticamente [111] y mecanismos de control de complejidad; y la tercera analizar la sensibilidad de las diferentes configuraciones, a los parámetros relacionados con el tamaño de la población, y el tamaño y composición del conjunto de funciones de análisis técnico utilizado.

## 6.2 Conjuntos de datos y diseño experimental

En esta sección se describe la configuración experimental utilizada, incluidos el conjunto de datos de entrenamiento y de prueba, el protocolo utilizado en la validación de los resultados del experimento, o la parametrización del algoritmo elegida; a lo que se acompañará de la presentación y discusión de los resultados obtenidos.

El estudio experimental se ha basado en la combinación de dos conjuntos de datos, que cubren los valores bursátiles de sesión diarios y los tipos de interés mensuales, de un período de doce años, del 2004 al 2015: el primero corresponde al índice bursátil Standard & Poor's 500, y el segundo a los datos de tipos de interés de letras del tesoro de la Reserva Federal Americana a tres meses, necesarios para calcular el rendimiento diario sin riesgo. El primero se obtuvo del proveedor comercial Datastream, y el segundo del Banco de la Reserva Federal de Atlanta, disponible<sup>1</sup> en la URL: <https://fred.stlouisfed.org/series/TB3MS>.

Como es sabido, las series financieras, a menudo, muestran cambios estructurales que afectan a su previsibilidad. Para hacer que el ejercicio de evaluación comparativa sea lo más genérico posible, se han probado las configuraciones basadas en la PG y las GE en tres escenarios experimentales diferentes. Para ello, los conjuntos de datos de doce años mencionados se han dividido en tres grupos de cuatro años consecutivos no solapados. Para cada uno de ellos, se han creado dos muestras diferentes: una para disponer de un conjunto de datos de entrenamiento, que cubre los primeros tres años, y otra muestra utilizada como datos de prueba, con el cuarto año restante. Esto ha dado como resultado los siguientes subconjuntos experimentales:

- 2004-2006 datos de entrenamiento, 2007 datos de prueba.
- 2008-2010 datos de entrenamiento, 2011 datos de prueba.
- 2012-2014 datos de entrenamiento, 2015 datos de prueba.

Para cada una de las muestras anteriores, se han probado las estrategias resultantes, de la combinación de las técnicas de computación evolutiva de la PG y

---

<sup>1</sup> Última comprobación de fecha 14/12/2018.

las GE, con el uso potencial de ADF y la aplicación del control del crecimiento, por medio de parsimonia paramétrica lexicográfica, a los individuos. Es decir, para cada muestra se ha realizado un estudio comparativo compuesto por los siguientes elementos:

- (PG). Programación genética básica.
- (PGPa). Programación genética usando parsimonia paramétrica lexicográfica.
- (PGAdf). Programación genética usando ADF.
- (PGAdfPA). Programación genética usando ADF y parsimonia paramétrica lexicográfica.
- (GE). Gramáticas evolutivas básicas.
- (GEPa). Gramáticas evolutivas usando parsimonia paramétrica lexicográfica.
- (GEAdf). Gramáticas evolutivas usando ADF
- (GEAdfPa). Gramáticas evolutivas usando ADF y parsimonia paramétrica lexicográfica.

El estudio experimental básico, se ha complementado con un análisis de sensibilidad paramétrica, para considerar el impacto de usar poblaciones más grandes e indicadores técnicos adicionales al conjunto básico utilizado. En este sentido han sido probadas las siguientes combinaciones de parámetros:

- 500 individuos.
- 3.000 individuos.
- 500 individuos más indicadores técnicos ampliados.
- 3.000 individuos más indicadores técnicos ampliados.

Como resultado de la aplicación del producto cartesiano, a las posibilidades de prueba mencionadas anteriormente, se han obtenido un total de 96 configuraciones diferentes a examinar. Dada la naturaleza estocástica de los algoritmos de CE, y en un esfuerzo por alcanzar un alto nivel de significación estadística, todos los experimentos se han repetido 2.000 veces. Esto suma un total de 192.000 experimentos realizados.

## 6.3 Resultados experimentales

En esta sección, se van a presentar los resultados de los experimentos realizados en tres apartados: el primer apartado, al que se considera el experimento base, se ha llevado a cabo con una población de 500 individuos y un conjunto de funciones reducido; el segundo apartado se corresponde con el análisis de sensibilidad paramétrica, que a su vez se ha dividido en tres partes: conjunto de funciones terminales extendido; población ampliada; y conjunto de funciones terminales y población ampliados.

### 6.3.1 Experimento base

El primer conjunto de experimentos, de este ejercicio de evaluación comparativa, establecerá la línea de base para los dos algoritmos de CE centrales y sus variantes paramétricas. Se comparará el rendimiento de la estrategia de PG y de las GE vs. B&H, utilizando poblaciones de 500 individuos y el conjunto de funciones terminales presentado en el capítulo 4. Los rendimientos de inversión compuestos, alcanzados en los resultados, se han calculado de acuerdo con las ecuaciones 4-7 y 4-9 que se describen en el apartado 4.3, y los retornos libres de riesgo conforme a la ecuación 4.3. Además de los retornos, se considerarán otros aspectos como la complejidad de las soluciones obtenidas o su confiabilidad.

La tabla 6.1, resume los principales resultados experimentales con respecto al rendimiento financiero en la muestra de prueba. Para cada configuración se muestra el rendimiento medio y la variación en las 2.000 ejecuciones efectuadas. Como es posible ver, las GE superan a la PG en los dos primeros períodos de tiempo (2007 y 2011). Esto viene acompañado de resultados más robustos, como lo demuestran las varianzas. Sin embargo, la superioridad mostrada por una técnica sobre otra se invierte en ambos frentes cuando se considera el último período final. También es posible comprobar, que la adición del mecanismo de control de parsimonia, al algoritmo base, generalmente produce mayores rendimientos en retorno de inversión, pero las ventajas de agregar la extensión de ADF son menos consistentes. Una razón puede ser debida a que las soluciones ideales en las que participan ADF, y en las que tienen más poder, son aquellas en las que los árboles de los individuos son de gran tamaño, con múltiples llamadas a subárboles dentro de ellos.



**Tabla 6.1:** Resumen de resultados del experimento base.

	Entrenamiento 2004-06		Entrenamiento 2008-10		Entrenamiento 2012-14	
	Resultados prueba 2007		Resultados prueba 2011		Resultados prueba 2015	
	Media	Varianza	Media	Varianza	Media	Varianza
<b>Ge</b>	0,0424	0,00057	0,0079	0,0018	-0,0155	0,00047
<b>GePa</b>	0,0432	0,00049	0,0124	0,0014	-0,0160	0,00055
<b>GeAdf</b>	0,0419	0,00039	-0,0081	0,0035	-0,0153	0,00041
<b>GeAdfPa</b>	0,0413	0,00028	-0,0040	0,0032	-0,0168	0,00058
<b>Gp</b>	0,0359	0,00077	-0,0633	0,0054	-0,0149	0,00028
<b>GpPa</b>	0,0366	0,00065	-0,0485	0,0051	-0,0148	0,00035
<b>GpAdf</b>	0,0367	0,00082	-0,0481	0,0051	-0,0154	0,00034
<b>GpAdfPa</b>	0,0363	0,00072	-0,0470	0,0048	-0,0154	0,00038
<b>B&amp;H</b>	0,0310		-0,0171		-0,0101	

(a) Retorno de inversión en muestras de prueba de 2.000 ejecuciones.

(b) En color azul claro se marcan las técnicas y períodos en los que se bate en rendimiento a B&H.

**Tabla 6.2:** Significación estadística del experimento base, de las diferencias en las muestras de prueba.

	Significación 2007 / 2011 / 2015							
	Ge	GePa	GeAdf	GeAdfPa	Gp	GpPa	GpAdf	GpAdfPa
<b>GePa</b>	-/--/=							
<b>GeAdf</b>	=/=/-	=/=/=						
<b>GeAdfPa</b>	++/=/=	++/=/=	=/--/=					
<b>Gp</b>	++/++/=	++/++/=	++/++/=	=/++/=				
<b>GpPa</b>	++/++/=	++/++/=	++/++/=	=/++/=	-/--/			
<b>GpAdf</b>	++/++/=	++/++/=	++/++/=	=/++/=	=/--/	=/=/		
<b>GpAdfPa</b>	++/++/=	++/++/=	++/++/=	=/++/=	=/--/	=/=/	=/=/	
<b>B&amp;H</b>	++/++/--	++/++/--	++/++/--	+/++/--	++/--/--	++/--/--	=/--/--	=/--/--

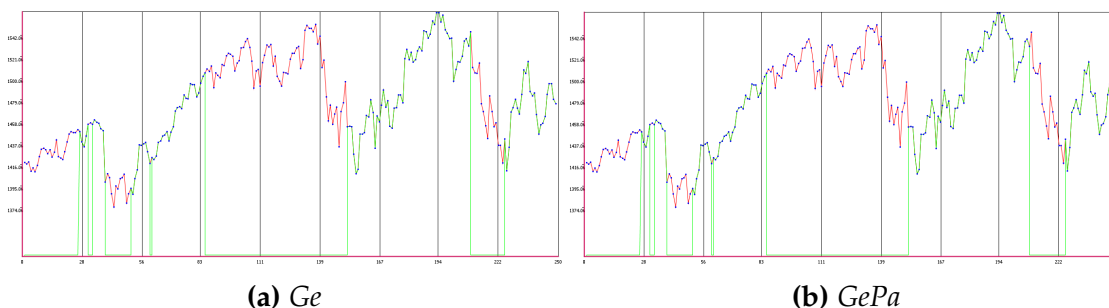
(a) En muestras de prueba de 2.000 ejecuciones.

Relacionado con lo expuesto anteriormente, la tabla 6.2 muestra la significación estadística de las diferencias de resultados ofrecidas en la tabla 6.1, para los tres períodos consecutivos. Cuando el algoritmo de una determinada columna ofre-

ce un rendimiento, de media/mediana, que es mayor que el de la fila con la que se compara en un nivel de significación de un 1 %, se usa el símbolo ++. Una diferencia similar en la dirección opuesta, se representa con --. En caso de que la diferencia sea significativa en el 5 %, nivel convencional, se usa - o +, según corresponda. Finalmente, si no es posible descartar la hipótesis nula de igualdad de los resultados se muestra =.

Las mejores estrategias por configuración y período, para experimento base, se presentan en el apéndice A.1. Para aquellas configuraciones que usan la técnica de ADF se proporcionan tanto la regla principal, el árbol "0" del individuo, como el árbol ADF, ADF1. En algunos casos, aunque el algoritmo de PG evolucionó el árbol de ADF, este no fue empleado por el árbol principal del individuo y la regla no los utilizó, por lo tanto falta la referencia a ADF1 en la expresión. El comportamiento de estas reglas se ilustra en apéndice A.2. La línea más clara que aparece en las figuras representa la recomendación de la regla. Si la recomendación es permanecer en el mercado esta línea sigue a la del índice (más oscura), por el contrario, si aconseja al inversor salir de él, la línea permanecerá plana en la parte inferior del gráfico. A continuación se reproducen dos de esas reglas, correspondientes al período *Entrenamiento 2004-2006. Prueba 2007* para las técnicas de las GE, Ge y GePa, figura 6.1.

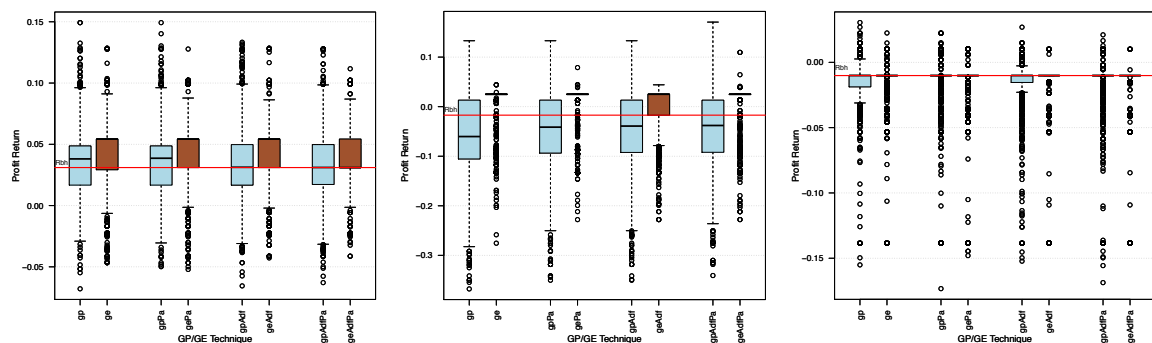
- *Ge*. Fitness: 0,12847  
Tree 0: (Or (>Mx2 M2) (Or (>Minimum UR) (<M10 LR)))
- *GePa*. Fitness: 0,12776  
Tree 0: (Or (>Mx2 M2) (<M10 LR))



**Figura 6.1:** Comportamiento de las mejores estrategias configuración línea base. Entrenamiento: 2004-2006, prueba: 2007. Ge y GePA.

La distribución de los rendimientos, por técnica y período, se muestra en la figura 6.2. En ella podemos ver tres paneles separados, que muestran diagramas de caja para los resultados de las diferentes configuraciones. Cada panel ofrece la distribución de los rendimientos en las muestras de prueba pertenecientes a los años 2007, 2011 y 2015, sobre 2.000 experimentos realizados. Como punto de referencia, se han agregado líneas horizontales de color rojo, que indican el rendimiento que se habría obtenido siguiendo a la estrategia de referencia de B&H. Es interesante señalar, que las distribuciones presentadas no son en absoluto simétricas, sino que tienden a ser sesgadas, mostrando colas más largas a la izquierda que representan estrategias deficientes que son muy poco rentables. Esto es especialmente evidente en el último período 6.2c que, como se puede ver en el correspondiente período de la tabla 6.1, es particularmente desafiante.

Figura 6.2: Distribución de resultados del experimento base por configuración.



(a) 2007 (Entrenamiento 2004-06). (b) 2011 (Entrenamiento 2008-10). (c) 2015 (Entrenamiento 2012-14).

(d) Muestras de prueba en 2.000 ejecuciones.

La tabla 6.3, ofrece información detallada sobre el rendimiento de las distintas configuraciones, en comparación con la estrategia de referencia B&H. Para cada una de ellas, se muestra el porcentaje de las estrategias de inversión que superaron en la prueba a B&H, columna de predicción “Mejor”, junto con el porcentaje de experimentos que resultan en estrategias que proporcionan básicamente el mismo rendimiento, columna de predicción nombrada como “Igual”. La última cifra, mostrada en la columna “Igual”, suele ser el resultado de estrategias que terminan adoptando la estrategia de referencia en la muestra de prueba. Este dato a menudo se pasa por alto en la literatura, pero es muy relevante porque proporciona información sobre la confiabilidad de los diferentes enfoques utilizados como generadores de estrategias de inversión rentables.

Los resultados presentados en la tabla 6.3, señalan que en el período 2004-2007 (prueba 2007) las GE obtienen reglas de inversión que igualan o baten a B&H entre un 74,65 % y un 90,10 % de los casos, mientras que la PG lo hace ligeramente por debajo del 54 % de las veces. Las GE obtiene mejores resultados en el período que comprende a 2008-2011 (prueba 2011), donde supera o mejora a B&H entre el 78,55 % y 90,50 % de los casos, mientras que la PG lo hace significativamente por debajo del 50 %, entre un 32,35 % y un 44,70 %. Cabe señalar que, tanto en el período comprendido entre 2004-2007, como en el período de 2008-2011, la versión estándar de las GE, y la versión con control de complejidad basada en parsimonia, muestran un rendimiento relativo especialmente bueno.

**Tabla 6.3:** Resultados del experimento base vs. Buy-and-Hold.

	2007 (Entrenamiento 2004-06)				2011 (Entrenamiento 2008-10)				2015 (Entrenamiento 2012-14)			
	Predicción		Complejidad		Predicción		Complejidad		Predicción		Complejidad	
	Mejor	Igual	Prof.	Nodos	Mejor	Igual	Prof.	Nodos	Mejor	Igual	Prof.	Nodos
<b>Ge</b>	63,70 %	10,95 %	3,24	7,56	84,80 %	2,20 %	2,89	5,81	7,95 %	70,45 %	3,66	8,68
<b>GePa</b>	65,45 %	14,10 %	2,97	6,38	88,85 %	1,65 %	2,79	5,31	4,55 %	79,45 %	3,19	6,88
<b>GeAdf</b>	55,05 %	31,25 %	2,96	6,09	74,70 %	0,85 %	2,74	5,11	0,50 %	87,65 %	2,89	5,48
<b>GeAdfPa</b>	49,75 %	40,35 %	2,73	5,06	77,50 %	1,05 %	2,67	4,86	0,30 %	87,30 %	2,87	5,33
<b>Gp</b>	53,60 %	0,35 %	6,68	32,02	28,90 %	3,45 %	6,85	36,29	12,45 %	50,25 %	6,17	24,29
<b>GpPa</b>	52,80 %	1,00 %	6,02	24,30	37,45 %	7,35 %	6,20	22,08	6,95 %	71,30 %	5,71	16,87
<b>GpAdf</b>	50,15 %	2,45 %	8,38	34,80	37,20 %	6,20 %	8,63	39,33	10,00 %	59,50 %	7,77	28,18
<b>GpAdfPa</b>	50,05 %	2,55 %	6,89	25,14	36,20 %	8,50 %	8,14	35,16	6,65 %	70,55 %	7,13	21,73
<b>Promedio GE</b>	58,49 %	24,16 %	2,98	6,27	81,46 %	1,44 %	2,77	5,27	3,33 %	81,21 %	3,15	6,59
<b>Promedio GP</b>	51,65 %	1,58 %	6,99	29,07	34,94 %	6,38 %	7,46	33,22	9,01 %	62,90 %	6,70	22,77

(a) Muestras de prueba en 2.000 ejecuciones.

(b) En color azul claro se marcan las técnicas y períodos en los que se bate en rendimiento a B&H.

En el lapso de tiempo comprendido entre los años 2012-2015 (prueba 2015), la situación es significativamente diferente. En ese período, las estrategias producidas por los dos algoritmos básicos generan muy pocas señales de compra o venta, y terminan efectivamente siguiendo una estrategia de B&H. Cuando ese no es el caso, tanto las GE como la PG son superadas por B&H con mucha más frecuencia que al contrario. Esto significa que, aunque los resultados promedio parecen ser muy bajos, esto se debe principalmente a la larga cola izquierda de la distribución de los rendimientos ya mencionados cuando describimos la figura 6.2. En la práctica, para todas las configuraciones excepto dos, la probabilidad de igualar a B&H se ha

movido entre un 70,45 % y un 87,65 %.

El segundo conjunto de columnas muestra la complejidad promedio de las soluciones obtenidas por las diferentes variantes del estudio. La primera columna indica la profundidad promedio de los árboles generados y la segunda el número total de nodos terminales y no terminales de los mismos. A primera vista parece, que la complejidad de las reglas es significativamente mayor cuando utilizamos la PG<sup>2</sup>, en vez de las GE, como algoritmo básico. Por ejemplo, en el período de 2008-2011 el número promedio de nodos para la PG es de 33,22 en comparación con los 5,27 de las GE. La profundidad promedio de las soluciones muestra un patrón similar ya que alcanza un 7,46 para la PG frente a un 2,77 para las GE. En cuanto al uso del operador de parsimonia (GpPa y GPAdfPa vs. GP y GPAdf), este reduce la complejidad de las soluciones basadas en la PG en 9,19 nodos y su profundidad en 0,57. Si consideramos las GE (GePa y GePaAdf vs. Ge y GeAdf), los efectos del mismo operador se mueven de forma idéntica en la misma dirección, pero estos tienden a ser más limitados porque ya es mucho menos complejo que el de la PG, por lo que se redujo el número promedio de nodos en 0,38 y la profundidad media en 0,09.

Con respecto al coste computacional, de este trabajo experimental, la complejidad computacional de las técnicas de las GE y la PG empleadas, depende de varios factores:

- Operadores genéticos aplicados (cruce, selección y mutación), que a su vez dependerán de las variantes e implementaciones de los mismos que se utilicen.
- Representación de los individuos y tamaño de la población.
- Número de generaciones evolucionadas.
- Número de experimentos realizados.
- Función de aptitud.

---

<sup>2</sup> Este comportamiento se puede explicar por la representación de los individuos. En la PG los individuos tienden a crecer constantemente, debido a las operaciones de cruce y mutación, si no se controla, en un proceso de crecimiento de código. Mientras que en las GE, tras varias generaciones, al crecer el número de individuos con un mejor valor de aptitud de promedio estos tienden a dominar la solución ignorándose así los efectos de crecimiento debidos al cruce y la mutación.

Dependiendo de la aplicación, aunque en general este es el caso, la función de aptitud suele ser el componente que soporta la carga computacional más intensa. En los experimentos realizados, la función de aptitud supera el coste computacional conjunto de los operadores genéticos, y de las operaciones de representación, significativamente. Como resultado, estas últimas pueden ignorarse y es por tanto posible caracterizar la complejidad del algoritmo como de orden lineal, en función del número de individuos, las generaciones y el número de experimentos realizados. En la forma dada por la ecuación 6-1.

$$O(jgn) \tag{6-1}$$

Donde:

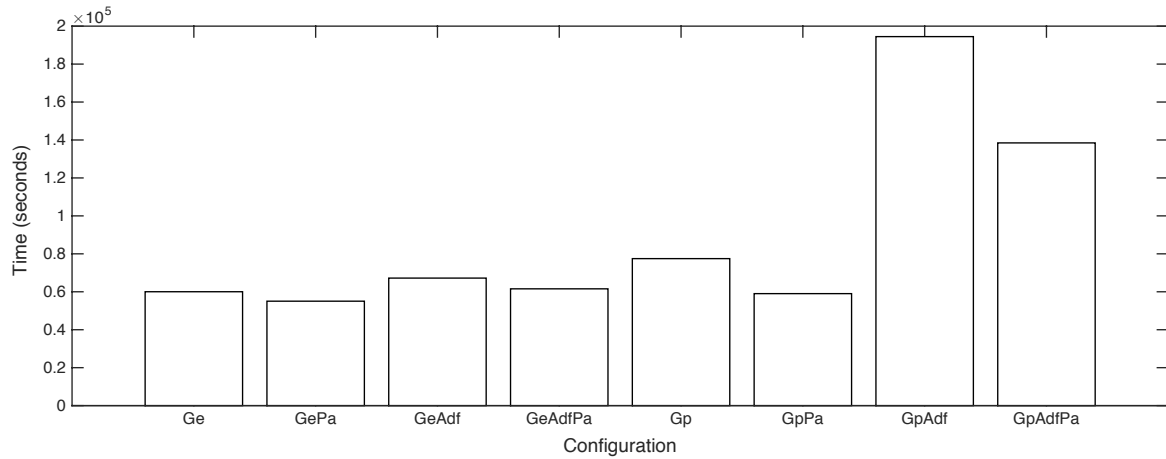
$g$  representa el número de generaciones.

$n$  el tamaño de la población.

$j$  el número de experimentos realizados.

La figura 6.3 resume, por configuración, el coste computacional total acumulado (en segundos) sobre las tres muestras. Es evidente que la versión estándar de la PG ha necesitado más tiempo de cálculo que las GE (un 29 % más). El uso de la parsimonia paramétrica lexicográfica redujo el coste computacional de ambas técnicas, pero la PG fue la que más se benefició de su empleo (24 % vs. 8 %). La diferencia entre los dos algoritmos básicos aumenta considerablemente una vez que se tiene en cuenta el uso de ADF. Las versiones basadas en la PG, que incluían esta característica, resultaron ser significativamente más costosas que las basadas en las GE.

Como resumen de esta parte del estudio, en la que se analizaban los algoritmos de CE, PG y GE, se puede concluir que, de acuerdo con los resultados obtenidos en los experimentos, las GE, particularmente en la versión estándar y con el control de parsimonia, han sido más robustas y han generado soluciones más simples que la PG. Al mismo tiempo las GE, han superado o igualado a la estrategia B&H significativamente más a menudo que la PG.

**Figura 6.3:** Coste computacional de los experimentos por configuración.

(a) Coste total acumulado expresado en segundos, sobre las tres muestras, por configuración.

(b) Los experimentos se ejecutaron en dos Intel I7-6500U de 2.5 Ghz con 16 Gbytes de RAM DDR3.

### 6.3.2 Análisis de sensibilidad paramétrica

Para enriquecer el análisis del experimento base, se ha probado la sensibilidad de los resultados aplicando dos extensiones: la primera aumentando el tamaño de la población; y la segunda aumentando el conjunto de indicadores técnicos que forman los elementos no terminales, ambos por separado y combinados. Esto ha definido tres conjuntos adicionales de experimentos cuyos resultados principales se resumen en la tabla 6.4.

El análisis de esta tabla 6.4 se tratará en las tres secciones siguientes, comparando sus resultados con la experimentación base. No obstante, en un primer análisis preliminar de esta tabla se puede apreciar, que los resultados son diversos. Se puede ver, que el aumento de los indicadores técnicos de los elementos no terminales ha supuesto una mejora en el 2007, para las técnicas de GE, siendo la mejora aun mayor para la PG, y que ambas siguen batiendo a B&H. En el 2011 y el 2015 ninguna es capaz de batir a B&H, empeorando sobre todo las GE. El aumento de la población, y la extensión de los indicadores técnicos junto con la población, supone una mejora para ambas técnicas en el 2007, sobre todo para la PG, que sobrepasan a las GE, no así en el 2011 y 2015 donde ambas empeoran, aunque en el año 2011 GE sigue batiendo a B&H. Es significativo observar, que el aumento de los indicadores técnicos, con o sin población extendida, ha resultado en un aumento significativo de los rendimientos de ambas técnicas en el año 2007.

**Tabla 6.4:** Análisis de sensibilidad al aumento del tamaño de la población y del conjunto de indicadores técnicos.

	Aumento ind. técnicos			Aumento de población			Aumento pob. e ind. Tec.		
	2007	2011	2015	2007	2011	2015	2007	2011	2015
<b>Ge</b>	0,04720**	-0,05556	-0,02056	0,03757**	-0,00790**	-0,01490	0,05495**	-0,01171**	-0,01605
<b>GePa</b>	0,04596**	-0,05745	-0,01986	0,03706**	0,00419**	-0,01567	0,05375**	-0,00344**	-0,01700
<b>GeAdf</b>	0,04651**	-0,07846	-0,02040	0,04131**	0,00409**	-0,01650	0,04967**	-0,00611**	-0,01927
<b>GeAdfPa</b>	0,04613**	-0,08342	-0,01917	0,04288**	0,01231**	-0,01667	0,04879**	-0,00293**	-0,01935
<b>Gp</b>	0,05671**	-0,06797	-0,01503	0,03779**	-0,07297	-0,01906	0,05992**	-0,06867	-0,01799
<b>GpPa</b>	0,05391**	-0,06631	-0,01501	0,03428**	-0,04932	-0,01654	0,05704**	-0,05537	-0,01812
<b>GpAdf</b>	0,05576**	-0,06192	-0,01559	0,04242**	-0,05781	-0,01754	0,06024**	-0,05685	-0,01490
<b>GpAdfPa</b>	0,05535**	-0,05679	-0,01500	0,04100**	-0,05727	-0,01659	0,06019**	-0,05784	-0,01438
<b>B&amp;H</b>	0,03100	-0,01710	-0,01010	0,03100	-0,01710	-0,01010	0,03100	-0,01710	-0,01010

\*\* Significación estadística vs. B&H al 1%. \*Significación estadística vs. B&H al 5%.

(a) Rendimientos promedio en muestras de prueba en 2.000 ejecuciones.

(b) En color azul claro se marcan las técnicas y períodos en los que se bate en rendimiento a B&H.

### 6.3.2.1 Conjunto de indicadores técnicos extendidos

Analizando los datos por separado, en este apartado, se ha probado la sensibilidad de los resultados al tamaño de indicadores técnicos aplicados. El objetivo era explorar, si los algoritmos podían explotar esta nueva información aportada para encontrar mejores reglas de inversión. Además de esto, se tenía la intención de ver si se posían encontrar patrones con respecto a las capacidades de las diferentes configuraciones para extraer rentabilidad adicional. Para este fin, se extendió la riqueza del conjunto de funciones básico sugerido por Lohpetch y Corne [120,121], manteniendo el tamaño de la población en 500 individuos y el resto de los parámetros de ejecución en los mismos valores que fueron usados en la experimentación base de referencia.

El conjunto extendido de funciones, que se ha incluido en este análisis, además de los indicadores ya utilizados en la experimentación de línea base, ha consistido en nuevos promedios móviles y puntos de pivote. Los puntos de pivote [164,208,216] se utilizan para determinar los soportes, las resistencias y los cambios de dirección en los precios de las acciones, y representan la posición del mercado en relación con sesiones anteriores. Además de los puntos de pivote estándar, que se



utilizarán en el análisis, existen otros muchos tipos como por ejemplo: los puntos de pivote de *Fibonacci*, los puntos de pivote de *Tom Demark*, los puntos de pivote de *Woodie* o los puntos de *Camarilla*. La elección de aquellos que se han utilizado en este estudio se ha basado simplemente en escoger los más habituales encontrados en la literatura.

Un punto pivote establece un cambio de niveles, o una condición, y consiste en un punto en el que el sentimiento de los inversores cambia de tendencia, pasando de tendencia alcista a tendencia bajista o viceversa. La naturaleza de estos indicadores tiene un carácter subjetivo, que se interpreta en la toma de decisiones de compra/venta de los modelos de inversión de la siguiente forma: si el mercado atraviesa este nivel desde abajo, se dice entonces que el sentimiento bursátil del inversor es de tipo alcista y se espera que el mercado continúe esa tendencia. Por el contrario, si este nivel es atravesado desde arriba, entonces el sentimiento bursátil es bajista y se confía que el mercado se mantenga en esa tendencia. También se supone que el punto pivote ofrece una cierta resistencia o soporte a la tendencia, y si el precio no puede romper este nivel entonces se vaticina un cambio de tendencia (rebote) o dirección en el corto plazo.

Los indicadores técnicos añadidos al conjunto de funciones han sido de dos tipos: los puntos pivote estándar <sup>3</sup> y una serie de medias móviles adicionales que se indican a continuación:

- Los indicadores de resistencia  $R1, R2, R3$  y de soporte  $S1, S2, S3$  se calculan como se expone a continuación:

El punto pivote es simplemente el promedio de la suma de los valores de una sesión bursátil siguientes: apertura, máximo, mínimo y cierre. Su cálculo sigue la fórmula 6-2<sup>4</sup>.

$$P = (O + H + L + C) / 4 \quad (6-2)$$

---

<sup>3</sup> No se han incluido en este experimento los puntos pivote de *Fibonacci*, *Demark* y *Woodie*. La elección de los indicadores técnicos adecuados forma parte de un estudio en si mismo que excede el objetivo de esta tesis.

<sup>4</sup> En otras implementaciones se usa  $(H + L + C) / 3$ , no teniéndose en cuenta el precio de apertura.

En donde:

$P$  es el punto pivote a calcular,

$O$  es el precio de apertura de sesión,

$H$  es el máximo valor alcanzado en el día,

$L$  es el mínimo valor de la sesión bursátil,

$C$  es el valor al cierre de sesión,

todos ellos referidos a la sesión previa.

El primer valor de soporte  $S1$  y de resistencia  $R1$  se obtienen de la observación del rango de precios por encima del punto pivote ( $2 \times P - H$ ), y por debajo de él ( $2 \times P - L$ ).

$$\begin{aligned} R1 &= P + (P - L) = 2 \times P - L \\ S1 &= P - (H - P) = 2 \times P - H \end{aligned} \quad (6-3)$$

- Los segundos valores de resistencia  $R2$  y de soporte  $S2$ , están por encima y por debajo de los valores de soporte y de resistencia anteriores  $R1$  y  $S1$ . Se calculan de la siguiente forma:

$$\begin{aligned} R2 &= P + (H - L) \\ S2 &= P - (H - L) \end{aligned} \quad (6-4)$$

- Y de igual manera se procede con los valores de la tercera resistencia y del tercer soporte, que se expresan en función de  $R1$  y  $S1$ , siguiendo las fórmulas siguientes:

$$\begin{aligned} R3 &= H + 2 \times (P - L) = R1 + H - L \\ S3 &= L - 2 \times (H - P) = S1 - H - L \end{aligned} \quad (6-5)$$

- Existen variantes de las fórmulas anteriores, en las que los niveles de soporte  $S$  y de resistencia  $R$  se calculan de distinta forma, variando ligeramente los resultados que se obtienen.
- Los indicadores técnicos  $M4$ ,  $M6$ ,  $M7$ ,  $M8$ ,  $M9$ ,  $M11$ ,  $M12$ , se corresponden con las medias móviles de precios de cierre durante un período de cuatro, seis, siete, ocho, nueve, once y doce meses anteriores al mes en curso, y se corresponden con 100, 150, 175, 200, 225, 275 y 300 sesiones, respectivamente.
- Otro de los puntos pivote, que permite pronosticar el futuro de una tendencia, es el punto pivote de *Demark*, mencionados anteriormente. Los puntos *Demark* no se consideran en realidad como puntos pivote, pero sirven para calcular los valores máximos y mínimos de un determinado período bursátil, y poder utilizar esta información como elemento de predicción. Para calcular estos niveles se emplean las fórmulas siguientes [173]:

$$\begin{aligned}
 \text{Si } C < O, & \implies X = H + (2 \times L) + C \\
 \text{Si } C > O, & \implies X = (2 \times H) + L + C \\
 \text{Si } C = O, & \implies X = H + L + (2 \times C)
 \end{aligned} \tag{6-6}$$

$$\begin{aligned}
 \text{Nuevo Maximo} &= (X/2) - L \\
 \text{Nuevo Minimo} &= (X/2) - H
 \end{aligned}$$

- Los puntos pivote de *Woodie* son similares a los puntos pivote clásicos, pero se calculan de forma diferente, otorgando una mayor importancia a los valores de precio de cierre de sesión del período bursátil anterior. Para obtener los puntos pivote de *Woodie* se emplean las fórmulas que se muestran a continuación [173]:

$$\begin{aligned}
 R2 &= P + H - L \\
 R1 &= (2 \times P) - L \\
 P &= (H + L + 2 \times C) / 4 \\
 S1 &= (2 \times P) - H \\
 S2 &= P - H + L
 \end{aligned} \tag{6-7}$$

- Los puntos pivote de *Camarilla* son utilizados por los operadores, para definir puntos de parada que limiten pérdidas o que señalen toma de beneficios. Sirven, en cualquier caso, para definir señales que aconsejen la salida del mercado. Constan de ocho niveles, cuatro de soporte y cuatro de resistencia, y se considera que proporcionan una alta probabilidad de comportarse como soportes y resistencias eficientes para controlar la tendencia actual. Para calcularlos se emplean las fórmulas siguientes [173]:

$$\begin{aligned}
 R4 &= (H - L) \times (1,1/2) + C \\
 R3 &= (H - L) \times (1,1/4) + C \\
 R2 &= (H - L) \times (1,1/6) + C \\
 R1 &= (H - L) \times (1,1/12) + C \\
 S1 &= C - (H - L) \times (1,1/12) \\
 S2 &= C - (H - L) \times (1,1/6) \\
 S3 &= C - (H - L) \times (1,1/4) \\
 S4 &= C - (H - L) \times (1,1/2)
 \end{aligned}
 \tag{6-8}$$

- Los pivotes de *Fibonacci* combinan a los pivotes tradicionales y a la secuencia de *Fibonacci* para determinar los niveles de resistencia y soporte en los cuales haya una alta probabilidad de que en el mercado se materialicen movimientos importantes. Para calcularlos se emplean las fórmulas siguientes [174]:

$$\begin{aligned}
 R3 &= P + 1,000 \times (H - L) \\
 R2 &= P + 0,618 \times (H - L) \\
 R1 &= P + 0,382 \times (H - L) \\
 P &= (H + L + C) / 3 \\
 S1 &= P - 0,382 \times (H - L) \\
 S2 &= P - 0,618 \times (H - L) \\
 S3 &= P - 1,000 \times (H - L)
 \end{aligned}
 \tag{6-9}$$

Si se comparan las cifras que se exponen en las tablas 6.1 y 6.4, con los de la tabla 6.5, es posible observar que hay una gran dispersión de cambios de dirección

en los resultados, dependiendo del período de tiempo elegido y del algoritmo básico considerado.

**Tabla 6.5:** Análisis de sensibilidad al aumento de indicadores técnicos vs. experimento base.

	Línea base			Conjunto extendido		
	2007	2011	2015	2007	2011	2015
<b>Ge</b>	0,0420**	0,0079**	-0,0155	0,0472**	-0,0556	-0.0206
<b>GePa</b>	0,0432**	0,0124**	-0,0160	0,0460**	-0,0574	-0.0199
<b>GeAdf</b>	0,0419**	-0,0081**	-0,0153	0,0465**	-0,0785	-0.0204
<b>GeAdfPa</b>	0,0413 *	-0,0040**	-0,0168	0,0461**	-0,0834	-0,0192
<b>Gp</b>	0,0359**	-0,0633	-0,0149	0,0567**	-0.0680	-0,01503
<b>GpPa</b>	0,0366**	-0,0485	-0,0148	0,0539**	-0.0663	-0,0150
<b>GpAdf</b>	0,0367	-0,0481	-0,0154	0,0558**	-0.0619	-0,0156
<b>GpAdfPa</b>	0,0363	-0,0470	-0,0154	0,0554**	-0.0568	-0,0150
<b>B&amp;H</b>	0,0310	-0,0171	-0,0101	0,0310	-0,0171	-0,0101

(a) Retorno de inversión de media, en muestras de prueba de 2.000 ejecuciones.

(b) En color azul claro se marcan las técnicas y períodos en los que se bate en rendimiento a B&H.

(c) \*\* Significación estadística vs. B&H al 1 %. \* Significación estadística vs. B&H al 5 %.

Para el conjunto de datos del período comprendido entre los años 2004-2007 (entrenamiento en el período 2004-2006, y prueba en el año 2007), agregar las nuevas funciones al conjunto de terminales da como resultado una mejora generalizada en los retornos de inversión. La escala del cambio es especialmente notable para las configuraciones basadas en el uso de la PG, que pasa a superar a las alternativas basadas en las GE en todos los casos.

Por otro lado, en la tabla 6.6 de resultados vs. B&H, se observa que la proporción de estrategias que fueron capaces de vencer a B&H en 2007 creció en todas las configuraciones de la PG utilizadas. Ese es también el caso de las configuraciones que descansan en las GE, que superan el 94 % de acierto cuando consideramos el número de técnicas que coinciden con la estrategia de referencia B&H. En el caso de la PG, a la que se aplica tanto el control de parsimonia lexicográfica como el uso de ADF, la proporción de estrategias que vencen a B&H alcanza un 73,95 %, por encima del resultado inicial de tan solo un 50,05 % de la estrategia base.

Para el resto de los períodos, los resultados alcanzados con la extensión del

**Tabla 6.6:** Resultados del aumento de indicadores técnicos vs. Buy-and-Hold.

	2007 (Entrenamiento 2004-06)				2011 (Entrenamiento 2008-10)				2015 (Entrenamiento 2012-14)			
	Predicción		Complejidad		Predicción		Complejidad		Predicción		Complejidad	
	Mejor	Igual	Prof.	Nodos	Mejor	Igual	Prof.	Nodos	Mejor	Igual	Prof.	Nodos
<b>Ge</b>	72,45%	21,50%	3,15	6,79	49,60%	1,00%	2,94	5,91	1,40%	80,00%	3,66	8,86
<b>GePa</b>	72,30%	23,40%	2,87	5,69	48,00%	1,25%	2,88	5,65	1,15%	82,40%	3,35	7,46
<b>GeAdf</b>	59,10%	38,00%	2,88	5,75	33,60%	1,20%	2,85	5,47	1,40%	81,30%	3,36	7,32
<b>GeAdfPa</b>	57,85%	36,90%	2,65	4,76	31,95%	0,90%	2,73	5,04	0,50%	85,45%	3,08	6,18
<b>Gp</b>	71,65%	0,40%	6,78	34,48	22,75%	12,30%	6,83	35,93	11,20%	53,10%	6,14	23,84
<b>GpPa</b>	71,30%	1,05%	6,54	30,36	22,25%	17,20%	6,74	33,49	7,05%	69,60%	5,70	16,91
<b>GpAdf</b>	71,20%	1,65%	8,04	34,57	27,20%	16,05%	8,35	36,69	2,10%	83,25%	7,84	30,69
<b>GpAdfPa</b>	73,95%	1,10%	7,21	29,20	28,20%	18,30%	7,77	32,42	5,90%	72,20%	7,31	22,61
<b>Promedio GE</b>	65,43%	29,95%	2,89	5,75	40,79%	1,09%	2,85	5,52	1,11%	82,29%	3,36	7,46
<b>Promedio GP</b>	72,03%	1,05%	7,14	32,15	25,18%	15,96%	7,42	34,63	6,56%	69,54%	6,75	23,15

(a) Muestras de prueba en 2.000 ejecuciones.

(b) En color azul claro se marcan las técnicas y períodos en los que se bate en rendimiento a B&H.

conjunto de terminales no son tan buenos. En el año 2011, ni las configuraciones basadas en la PG, ni las basadas en las GE, pueden superar a la estrategia de referencia B&H en términos de rendimiento promedio. Parece que hay una disminución general en el rendimiento que afecta a las GE en mayor medida. Lo mismo se aplica al año 2015. Es interesante observar, que para estos años se aprecia una fuerte disminución en el número de estrategias que superan a B&H, que es especialmente aguda en el año 2011 para las GE (del orden del 30% al 40%, dependiendo de la configuración), mientras que la proporción de estrategias que siguen al valor de referencia, dependiendo de las configuraciones pero sobre todo en la PG, aumenta. Una posible explicación, para este comportamiento, sería el hecho de que al aumentar el número de indicadores se expande el espacio de la solución. Dado que se mantuvo constante el tamaño de la población, y el número de generaciones, los algoritmos podrían no haber convergido en las mejores soluciones. Eso podría explicar por qué no pudieron explotar convenientemente la nueva información aportada.

Al igual que en la prueba anterior, la complejidad y profundidad de los individuos obtenidos por la PG es significativamente mayor que la alcanzada por las soluciones basadas en las GE. La complejidad promedio de las soluciones generadas por la PG y las GE es bastante similar a la calculada para las configuraciones

del experimento base, tanto en número de nodos como en la profundidad del árbol construido.

### 6.3.2.2 Población extendida

En esta prueba de sensibilidad paramétrica, la modificación introducida consistió en aumentar el número de individuos de la población de 500 a 3.000. Los resultados, como se esperaba, muestran signos claros de un sobreentrenamiento. Todas las configuraciones basadas en la PG y las GE obtienen mayores retornos de inversión en el entrenamiento. Sin embargo, una vez que es considerada la muestra de prueba, el rendimiento no es tan bueno.

**Tabla 6.7:** *Análisis de sensibilidad al aumento de la población vs. experimento base.*

	Línea base			Población extendida		
	2007	2011	2015	2007	2011	2015
<b>Ge</b>	0,0420**	0,0079**	-0,0155	0,0376**	-0,0079**	-0,0149
<b>GePa</b>	0,0432**	0,0124**	-0,0160	0,0371**	0,0042**	-0,0157
<b>GeAdf</b>	0,0419**	-0,0081**	-0,0153	0,0413**	0,0041**	-0,0165
<b>GeAdfPa</b>	0,0413 *	-0,0040**	-0,0168	0,0429**	0,0123**	-0,0167
<b>Gp</b>	0,0359**	-0,0633	-0,0149	0,0378**	-0,0730	-0,0191
<b>GpPa</b>	0,0366**	-0,0485	-0,0148	0,0343**	-0,0493	-0,0165
<b>GpAdf</b>	0,0367	-0,0481	-0,0154	0,0424**	-0,0578	-0,0175
<b>GpAdfPa</b>	0,0363	-0,0470	-0,0154	0,0410**	-0,0573	-0,0166
<b>B&amp;H</b>	0,0310	-0,0171	-0,0101	0,0310	-0,0171	-0,0101

(a) *Retorno de inversión de media, en muestras de prueba de 2.000 ejecuciones.*

(b) *En color azul claro se marcan las técnicas y períodos en los que se bate en rendimiento a B&H.*

(c) *\*\* Significación estadística vs. B&H al 1 %. \* Significación estadística vs. B&H al 5 %.*

En los resultados, figuras 6.7, se ha observado que la predicción de los algoritmos en la fase de prueba experimenta diversas variaciones: por ejemplo, en la muestra de prueba del año 2007 ha empeorado para el algoritmo de GE en las técnicas (Ge y GePa) y permanecido invariable para las técnicas que usaron ADF, por otro lado, ha mejorado ligeramente para las configuraciones de la PG que utilizan la técnica de ADF (GpAdf y GpAdfPa). En el año 2011, la mayoría de las configuraciones obtuvieron peores resultados, solo mejoraron los enfoques basados en

las GE en los que se aplicó el uso de ADF (GeAdf y GeAdfPa). Dicho esto, las variaciones en términos absolutos son muy pequeñas. Finalmente, en el año 2015, la GE permanece casi igual, pero la PG se ve afectada negativamente. En cuanto al rendimiento de ambas soluciones. frente a la referencia B&H, las evidencias se mezclan, una vez más, en el sentido de que la PG y las GE también lo superaron en el período 2004-2007; solo las variaciones relacionadas con la GE lo hacen en el período 2008-2011, y tanto PG como la GE no lo consiguen en el período 2012-2015.

En cuanto a la complejidad y la profundidad de los individuos, tabla 6.8, obtenidas por la PG, estas son significativamente mayores que las alcanzadas por las GE. Por ejemplo en el período 2004-2007: un 7,72 de la PG frente a un 3,92 de las GE en profundidad y 38,82 de la PG contra un 9,96 de las GE en complejidad. Al comparar los resultados con el experimento de línea base de la tabla 6.3, la complejidad de la solución lograda tanto en la PG como en las GE aumenta con el número de individuos, y obtenemos los siguientes resultados: en la PG el número de nodos aumenta entre un 6,02 y un 6,99, y de un 1,89 a un 3,11 en las GE. Si consideramos la profundidad, el valor promedio para la PG crece de un 1,54 para las versiones sin control de parsimonia lexicográfica, contra un 0,61 para aquellas que lo implementan. El impacto de la complejidad en las GE es similar, ya que estos aumentos son de un 0,79 y un 0,49 respectivamente.

**Tabla 6.8:** Resultados del aumento de la población vs. Buy-and-Hold.

	2007 (Entrenamiento 2004-06)				2011 (Entrenamiento 2008-10)				2015 (Entrenamiento 2012-14)			
	Predicción		Complejidad		Predicción		Complejidad		Predicción		Complejidad	
	Mejor	Igual	Prof.	Nodos	Mejor	Igual	Prof.	Nodos	Mejor	Igual	Prof.	Nodos
<b>Ge</b>	43,05 %	1,90 %	4,47	12,50	68,40 %	5,60 %	4,29	11,63	7,20 %	72,40 %	3,66	8,61
<b>GePa</b>	44,40 %	1,80 %	4,07	10,64	80,75 %	2,50 %	3,53	8,33	3,55 %	80,75 %	3,16	6,67
<b>GeAdf</b>	60,95 %	1,50 %	3,81	9,22	80,50 %	3,15 %	3,66	8,45	3,15 %	79,30 %	3,24	7,00
<b>GeAdfPa</b>	66,05 %	1,55 %	3,32	7,47	88,80 %	1,50 %	3,05	6,16	1,25 %	84,30 %	3,00	5,86
<b>Gp</b>	55,25 %	0,00 %	6,86	37,21	23,55 %	0,00 %	6,92	40,52	11,80 %	28,25 %	6,75	32,57
<b>GpPa</b>	51,65 %	0,00 %	6,71	34,13	35,50 %	0,50 %	6,22	23,12	14,60 %	42,30 %	5,95	21,69
<b>GpAdf</b>	60,10 %	0,35 %	9,09	42,05	33,75 %	0,80 %	9,46	46,58	11,40 %	36,50 %	8,75	37,90
<b>GpAdfPa</b>	59,45 %	0,10 %	8,23	33,89	34,00 %	1,10 %	8,96	42,05	12,80 %	44,95 %	7,65	27,57
<b>Promedio GE</b>	53,61 %	1,69 %	3,92	9,96	79,61 %	3,19 %	3,63	8,64	3,79 %	79,19 %	3,27	7,04
<b>Promedio GP</b>	56,61 %	0,11 %	7,72	36,82	31,70 %	0,60 %	7,89	38,07	12,65 %	38,00 %	7,28	29,93

(a) Muestras de prueba en 2.000 ejecuciones.

(b) En color azul claro se marcan las técnicas y períodos en los que se bate en rendimiento a B&H.



En resumen, las GE siguen brindando, en los experimentos realizados, una aparente mayor fiabilidad en términos de probabilidad de vencer a B&H, ya que lo hacen en dos de las tres muestras usadas, y también genera soluciones más simples. El uso de ADF y el mecanismo de control de parsimonia mejora los resultados, y el mero aumento del tamaño de la población, generalmente, no resulta en un mejor desempeño.

### 6.3.2.3 Conjunto de indicadores técnicos y población extendidos

En este caso, se ha aumentado tanto la población como el número de indicadores técnicos utilizados. Aparte de eso, se han mantenido el resto de los parámetros de ejecución del experimento base constantes.

La combinación de estos cambios no ha tenido ningún impacto importante en términos relativos. En general, el rendimiento promedio de las estrategias de inversión sufre una leve degradación en la mayoría de las configuraciones aplicadas. Dicho esto, vale la pena señalar que las estrategias basadas en las GE y sobre todo en la PG se beneficiaron especialmente, de estos cambios introducidos, en la muestra de prueba del año 2007. Véase tabla 6.9.

**Tabla 6.9:** *Análisis de sensibilidad al aumento de la población e indicadores técnicos vs. experimento base.*

	Línea base			Población-Funciones extendidas		
	2007	2011	2015	2007	2011	2015
<b>Ge</b>	0.0420**	0.0079**	-0.0155	0.0550**	-0.0117**	-0.0161
<b>GePa</b>	0.0432**	0.0124**	-0.0160	0.0538**	-0.0034**	-0.0170
<b>GeAdf</b>	0.0419**	-0.0081**	-0.0153	0.0497**	-0.0061**	-0.0193
<b>GeAdfPa</b>	0.0413 *	-0.0040**	-0.0168	0.0488**	-0.0029**	-0.0194
<b>Gp</b>	0.0359**	-0.0633	-0.0149	0.0599**	-0.0687	-0.0180
<b>GpPa</b>	0.0366**	-0.0485	-0.0148	0.0570**	-0.0554	-0.0181
<b>GpAdf</b>	0.0367	-0.0481	-0.0154	0.0602**	-0.0569	-0.0149
<b>GpAdfPa</b>	0.0363	-0.0470	-0.0154	0.0602**	-0.0578	-0.0144
<b>B&amp;H</b>	0.0310	-0.0171	-0.0101	0.0310	-0.0171	-0.0101

(a) Retorno de inversión de media, en muestras de prueba de 2.000 ejecuciones.

(b) En color azul claro se marcan las técnicas y períodos en los que se bate en rendimiento a B&H.

(c) \*\* Significación estadística vs. B&H al 1 %. \* Significación estadística vs. B&H al 5 %.

Como es posible apreciar en la tabla 6.10, cuando en las configuraciones consideramos la proporción de reglas encontradas que vencieron o básicamente replicaron a la estrategia de referencia B&H, una vez más se ve la importancia de la distribución de los resultados obtenidos. La proporción de estrategias que replicaron o superaron a B&H aumentó del 67,94 % de partida al 83,09 % (91,19 % las GE y 75 % la PG) en el año 2007; bajó ligeramente del 62,11 % al 57,53 % (79,23 % las GE y 35,83 % la PG) en el 2011 y subió de nuevo del 78,23 % al 85,98 % (84,70 % las GE y 87,26 % la PG) en el 2015. La solidez de las estrategias mejora en dos de las tres muestras utilizadas, pero los resultados de rendimiento promedio se ven arrastrados por algunas estrategias débiles que ofrecieron comportamientos muy pobres.

Al igual que en las tres configuraciones experimentales anteriores, la profundidad promedio y el número de nodos de los individuos obtenidos por las configuraciones basadas en la PG, en los tres períodos, son claramente mayores que los obtenidos por el conjunto de soluciones que utilizaron las GE como algoritmo central: un 7,74, comparado con un 3,69 en valores de profundidad del árbol y un 8,88 en comparación con un 35,66 en la complejidad. Finalmente, la complejidad de las soluciones logradas por la PG y las GE son muy similares a las observadas en la prueba con una población de 3.000 individuos.

**Tabla 6.10:** Resultados del aumento de población e indicadores técnicos vs. Buy-and-Hold.

	2007 (Entrenamiento 2004-06)				2011 (Entrenamiento 2008-10)				2015 (Entrenamiento 2012-14)			
	Predicción		Complejidad		Predicción		Complejidad		Predicción		Complejidad	
	Mejor	Igual	Prof.	Nodos	Mejor	Igual	Prof.	Nodos	Mejor	Igual	Prof.	Nodos
<b>Ge</b>	87,85 %	0,50 %	3,89	9,73	73,50 %	2,30 %	3,51	8,39	1,85 %	84,70 %	4,62	13,00
<b>GePa</b>	89,95 %	0,15 %	3,64	8,49	78,80 %	1,60 %	3,15	6,88	1,05 %	85,55 %	4,17	10,85
<b>GeAdf</b>	88,60 %	4,30 %	3,86	7,57	77,70 %	1,65 %	3,14	6,59	2,05 %	79,10 %	4,63	12,16
<b>GeAdfPa</b>	89,70 %	4,15 %	3,10	6,20	80,65 %	0,70 %	2,85	5,38	1,20 %	83,30 %	3,70	11,34
<b>Gp</b>	76,60 %	0,00 %	6,85	38,24	23,05 %	3,50 %	6,89	39,62	1,35 %	87,65 %	8,46	35,37
<b>GpPa</b>	72,55 %	0,00 %	6,82	37,51	27,50 %	6,70 %	6,21	23,00	1,25 %	80,60 %	6,48	29,67
<b>GpAdf</b>	76,10 %	0,00 %	8,77	40,67	29,50 %	11,70 %	8,89	41,28	1,35 %	87,65 %	8,46	35,37
<b>GpAdfPa</b>	74,75 %	0,00 %	8,36	37,24	26,45 %	14,90 %	8,54	38,53	0,80 %	88,40 %	8,12	31,36
<b>Promedio GE</b>	89,03 %	2,16 %	3,62	8,00	77,66 %	1,56 %	3,16	6,81	1,54 %	83,16 %	4,28	11,84
<b>Promedio GP</b>	75,00 %	0,00 %	7,70	38,42	26,63 %	9,20 %	7,63	35,61	1,19 %	86,08 %	7,88	32,94

(a) Muestras de prueba en 2.000 ejecuciones.

(b) En color azul claro se marcan las técnicas y períodos en los que se bate en rendimiento a B&H.

La tabla 6.11, resume los principales resultados del ejercicio de evaluación comparativa. Allí se presentan para todas las variantes y parámetros de configuraciones iniciales de los experimentos, un resumen de las métricas consideradas en 2.000 ejecuciones en cada una de las tres muestras de prueba. Específicamente, se expone la proporción de reglas de inversión que proporcionan rendimientos iguales o mejores a la estrategia de referencia B&H, junto con sus valores de complejidad alcanzados en términos de profundidad y número de nodos del árbol de los individuos.

**Tabla 6.11:** Resumen del análisis de sensibilidad.

	Configuración			Aumento			Aumento de			Aumento de		
	Línea base			Ind. técnicos			Población			Pop. e ind. Tec.		
	Ret. >=	Complejidad		Ret. >=	Complejidad		Ret. >=	Complejidad		Ret. >=	Complejidad	
B&H	Profund.	Nodos	B&H	Profund.	Nodos	B&H	Profund.	Nodos	B&H	Profund.	Nodos	
<b>Ge</b>	80.02%	3.26	7.35	75.32%	3.25	7.19	66.18%	4.14	10.91	83.42%	4.01	10.37
<b>GePa</b>	84.68%	2.98	6.19	76.17%	3.03	6.27	71.25%	3.59	8.55	85.70%	3.65	8.74
<b>GeAdf</b>	83.33%	2.86	5.56	71.53%	3.03	6.18	76.18%	3.57	8.22	84.47%	3.88	8.77
<b>GeAdfPa</b>	85.42%	2.76	5.08	71.18%	2.82	5.33	81.15%	3.12	6.50	86.57%	3.22	7.64
<b>Gp</b>	49.66%	6.57	30.87	57.13%	6.58	31.42	39.62%	6.84	36.77	64.05%	7.40	37.74
<b>GpPa</b>	58.95%	5.98	21.08	62.92%	6.33	26.92	48.18%	6.29	26.31	62.87%	6.50	30.06
<b>GpAdf</b>	55.17%	8.26	34.10	67.15%	8.08	33.97	47.63%	9.10	42.18	68.77%	8.71	39.11
<b>GpAdfPa</b>	58.17%	7.39	27.34	66.55%	7.43	28.08	50.80%	8.28	34.50	68.43%	8.34	35.71
<b>Promedio GE</b>	83,36%	2,97	6,05	73,55%	3,03	6,24	73,69%	3,61	8,55	85,04%	3,69	8,88
<b>Promedio GP</b>	55,49%	7,05	28,35	63,44%	7,10	30,10	46,56%	7,63	34,94	66,03%	7,74	35,66
Promedios Totales												
	Ret. >=	Complejidad										
	B&H	Profund.	Nodos									
<b>Promedio GE</b>	78,91%	3,32	7,43									
<b>Promedio GP</b>	57,88%	7,38	32,26									

(a) Proporción de reglas de inversión que proporcionan rendimientos iguales o mejores que B&H junto con su complejidad.

(b) Promedios en 2.000 ejecuciones y en las tres muestras de prueba.

Si se consideran todos los experimentos y configuraciones realizados, tabla 6.12, los dos algoritmos centrales ofrecen un rendimiento promedio muy similar. Dicho esto, las GE con un 0,34% vencen a la PG en un 1,24% en términos absolutos (la PG muestra unas pérdidas absolutas de 0,9%). La evolución gramatical también resultó ser más fiable. Se puede ver que las configuraciones basadas en las GE obtuvieron resultados iguales o mejores que B&H en un 78,91% de los experimentos. Mientras que la programación genética igualó o mejoró el rendimiento de

B&H en tan solo un 57,88 % de las veces. En cuanto a la estructura de las reglas de inversión, también esta difirió entre las diferentes técnicas. Las GE proporcionaron individuos más simples en términos de complejidad: si observamos la profundidad promedio de los árboles de los individuos, se pasa de 3,32 de las GE contra 7,38 de la PG; y el número de nodos también es inferior, de 7,43 de las GE a 32,26 de la PG.

**Tabla 6.12:** Promedios de rentabilidad vs. B&H de todas las configuraciones.

	Configuración			Aumento			Aumento de			Aumento de			Media
	Línea base			Ind. técnicos			Población			Pop. e ind. Tec.			
	2007	2011	2015	2007	2011	2015	2007	2011	2015	2007	2011	2015	
Ge	0,0424**	0,0079**	-0,0155	0,0472**	-0,0556	-0,0206	0,0376**	-0,0079**	-0,0149	0,0550**	-0,0117**	-0,0162	0,0040
GePa	0,0432**	0,0124**	-0,0160	0,0460**	-0,0575	-0,0199	0,0371**	0,0042**	-0,0157	0,0538**	-0,0034**	-0,0170	0,0056
GeAdf	0,0419**	-0,0081**	-0,0153	0,0465**	-0,0785	-0,0204	0,0413**	0,0041**	-0,0165	0,0497**	-0,0061**	-0,0193	0,0016
GeAdfPa	0,0413**	-0,0040**	-0,0168	0,0461**	-0,0834	-0,0192	0,0429**	0,0123**	-0,0167	0,0488**	-0,0029**	-0,0194	0,0024
Gp	0,0359**	-0,0633	-0,0149	0,0567**	-0,0680	-0,0150	0,0378**	-0,0730	-0,0191	0,0599**	-0,0687	-0,0180	-0,0125
GpPa	0,0366**	-0,0485	-0,0148	0,0539**	-0,0663	-0,0150	0,0343**	-0,0493	-0,0165	0,0570**	-0,0554	-0,0181	-0,0085
GpAdf	0,0367**	-0,0481	-0,0154	0,0558**	-0,0619	-0,0156	0,0424**	-0,0578	-0,0175	0,0602**	-0,0569	-0,0149	-0,0078
GpAdfPa	0,0363**	-0,0470	-0,0154	0,0554**	-0,0568	-0,0150	0,0410**	-0,0573	-0,0166	0,0602**	-0,0578	-0,0144	-0,0072
Promedio GE	0,0422	0,0021	-0,0159	0,0464	-0,0687	-0,0200	0,0397	0,0032	-0,0159	0,0518	-0,0061	-0,0179	<b>0,0034</b>
Promedio GP	0,0364	-0,0517	-0,0151	0,0554	-0,0633	-0,0152	0,0391	-0,0593	-0,0174	0,0594	-0,0597	-0,0164	<b>-0,0090</b>
B&H	0,0310	-0,0171	-0,0101	0,0310	-0,0171	-0,0101	0,0310	-0,0171	-0,0101	0,0310	-0,0171	-0,0101	<b>0,0013</b>

\*\* Significación estadística vs. B&H al 1 %. \*Significación estadística vs. B&H al 5 %.

(a) Rendimientos promedio en muestras de prueba en 2.000 ejecuciones.

(b) En color azul claro se marcan las técnicas y períodos en los que se bate en rendimiento a B&H.

Con respecto a los beneficios de implementar la extensión de parsimonia paramétrica lexicográfica, o el uso de las ADF, la evidencia no es clara en términos de rendimientos, pero si lo es si consideramos la solidez de las soluciones. Las configuraciones que utilizaron el mecanismo de control de complejidad ofrecieron mejores rendimientos promedio que las que no lo hicieron. Aunque eso se aplica tanto a las GE como a la PG, esta última se benefició más de ello. El impacto de las ADF fue limitado en términos absolutos. Sin embargo, a pesar de ser un valor añadido para la PG, resultó tener un impacto negativo en las GE, reduciendo el rendimiento promedio de la inversión de un 0,48 % a un 0,20 %. Si consideramos la probabilidad de obtener resultados que sean al menos tan buenos como B&H, ambos mecanismos proporcionan valor añadido a las estrategias. Curiosamente, la contribución promedio que se obtuvo en nuestros experimentos fue dispar. El empleo de la parsimonia mejora la distribución de las ganancias promedio, de todas las configuraciones de las GE, pero el uso de ADF las empeora, mientras que tanto

el uso de parsimonia como el de ADF las mejora en promedio en la PG. Podemos decir, que las ADF contribuyen mucho más en los resultados ofrecidos por la PG<sup>5</sup>, la robustez aumenta desde un 55,42 % para las configuraciones sin ADF (promedio de gp y gpPa) hasta un 60,33 % (promedio de gpAdf y gpAdfPa), y por otra parte como hemos visto tanto la técnica de parsimonia como la de las ADF contribuyen a obtener mejores rendimientos en la PG, mientras solo la parsimonia lo hace en las GE.

El análisis de los resultados experimentales, en relación con el impacto diferencial, que podría tener el aumento en el tamaño de la población o el conjunto de terminales según los algoritmos centrales, nos lleva a las siguientes consideraciones. Si nos centramos en la robustez, la extensión de la población fue negativa tanto para las GE como para la PG. La proporción de reglas basadas en la PG, que ofrecieron rendimientos iguales o superiores a B&H, sufrió una caída absoluta muy parecida a la de las GE, 8,93 % de la PG frente a una caída del 9,67 % para las GE. Este efecto difiere con lo observado cuando consideramos los rendimientos promedio. Si bien este indicador permanece estable para la PG, las GE se ven afectadas más negativamente, lo que aumenta la importancia de las distribuciones y la probabilidad de obtener estrategias especialmente malas.

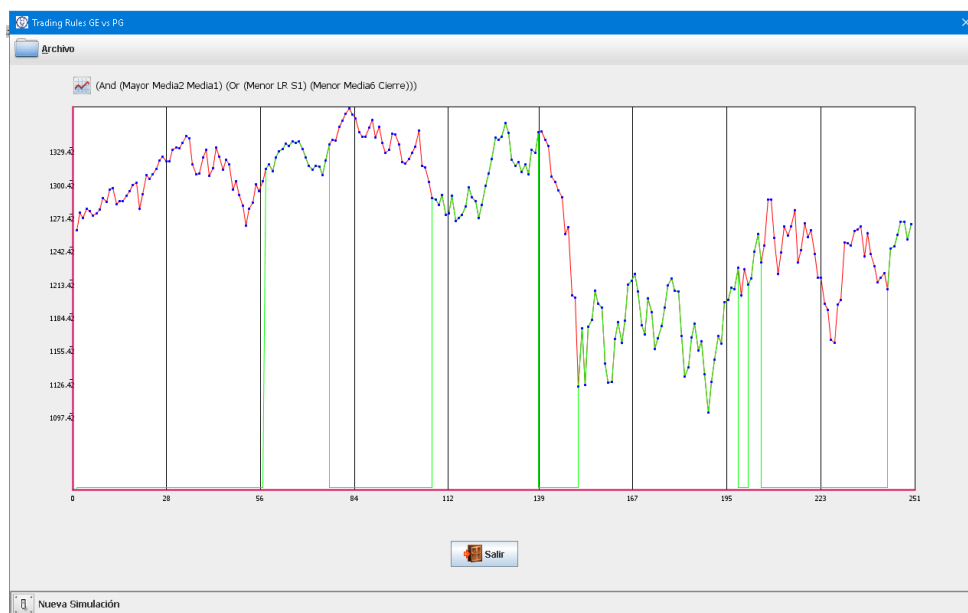
La extensión del conjunto de terminales hace evidente el mismo fenómeno que encontramos para el tamaño de la población. Por un lado, hay una divergencia entre el sentido del cambio en el rendimiento promedio y la solidez, y por otro el impacto del cambio en las dos técnicas es muy diferente. Si comparamos los resultados del caso base con la población extendida, con el conjunto de funciones extendidas y con ambas extensiones, vemos que la adición de los puntos de pivote y los nuevos promedios móviles ayudaron a las GE y la PG a mejorar el promedio de la solidez de las estrategias de inversión. La influencia en las GE, en la predicción, es limitada pasando de un 83,36 % a un 85,04 %, pero fue muy positiva para la PG ya que aumentó del 55,49 % al 63,44 %. Con respecto a los retornos promedio, la mejora de 0,25 % en términos absolutos que vemos en la PG viene acompañada de una caída de 0,05 % para las GE. Lo que de nuevo pone de manifiesto el impacto de las distribuciones en la probabilidad de contar con reglas catastróficas.

---

<sup>5</sup> Como ya se comentó para el experimento base, las soluciones ideales en las que las ADF son más eficientes son aquellas en las que los árboles de los individuos son de gran tamaño, con múltiples llamadas a subárboles dentro de ellos, y en este caso los árboles de la PG parecen adaptarse más a esas características.

Una última consideración sobre los rendimientos obtenidos en el 2011 por las GE. En ese lapso de tiempo, las GE son capaces de conseguir rendimientos positivos de 1,24 %, en un mercado bajista, que terminó el año con pérdidas de un 1,17 %. Y ello a pesar de la crisis del techo de deuda del 2011 en EEUU, que supuso, para el mercado de valores, un período de grandes tensiones en los períodos previos o posteriores al debate sobre el límite de la deuda de 2011. Por ejemplo, el promedio industrial Dow Jones sumó pérdidas de aproximadamente unos 2.000 puntos, desde los últimos días de julio hasta los primeros días de agosto. Incluso el Dow Jones registró una de sus peores caídas, en un solo día en su registro histórico, el 8 de agosto del 2011, el día después de la baja de S&P, que llegó a caer 635 puntos. En estas circunstancias, las GE fueron capaces de detectar los movimientos del mercado, capturando el punto en el que debía abandonarlo. En la figura 6.4, se muestra una solución del algoritmo básico con parsimonia “gePa”, que obtuvo un 14,37 % de rendimiento, y otra sin parsimonia “ge” en la figura 6.5, que obtuvo un 15,22 % de rendimiento. Se puede apreciar como en la sesión 139, que se corresponde con finales de julio del 2011, ambas soluciones salen del mercado y se posicionan en efectivo.

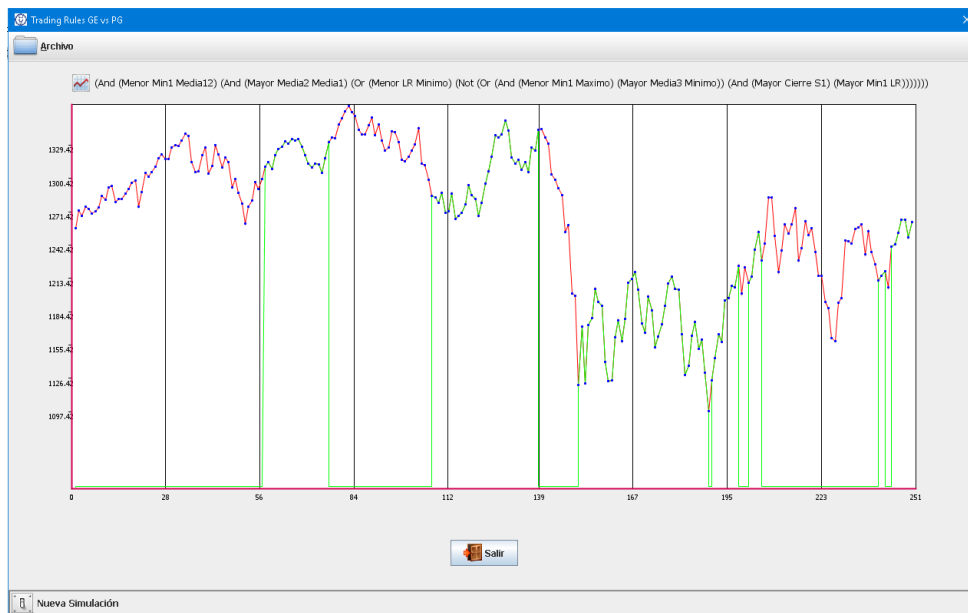
**Figura 6.4:** Comportamiento de la regla “gePa” (And (Mayor Media2 Media1) (Or (Menor LR S1) (Menor Media6 Cierre))) en 2011.



(a) La línea verde representa cuando la regla sugiere estar en el mercado, o cuando recomienda permanecer en efectivo.

En las figuras 6.4, 6.5 también se puede apreciar la diferencia en la complejidad de las soluciones, aquella que hizo uso de parsimonia es mucho más sencilla de interpretar.

**Figura 6.5:** Comportamiento de la regla "ge" (And (Menor Min1 Media12) (And (Mayor Media2 Media1) (Or (Menor LR Minimo) (Not (Or (And (Menor Min1 Maximo) (Mayor Media3 Minimo)) (And (Mayor Cierre S1) (Mayor Min1 LR)))))) en 2011.



(a) La línea verde representa cuando la regla sugiere estar en el mercado, o cuando recomienda permanecer inactivo.

## 6.4 Resumen y conclusiones

El uso de la computación evolutiva, en el dominio de la generación y optimización de estrategias de inversión, tiene una larga historia. Una de las líneas de investigación más populares es la que se ha centrado en el uso de las técnicas de PG y GE, para desarrollar reglas de inversión basadas en indicadores técnicos. Entre las principales ventajas de este enfoque, podemos destacar la posibilidad de obtener reglas que sean flexibles, ya que la estructura no está completamente predefinida de antemano, y que estas sean simples e interpretables.

La literatura que versa sobre este tema está basada en estudios con configuraciones muy diferentes. Algunos autores, la mayoría, confían en el uso de la PG y

otros en las GE; algunos consideran además extensiones como la adición de mecanismos de control de parsimonia o la implementación de técnicas de ADF, mientras que otros no. Dado que la mayoría de los estudios también se basan en muestras de series temporales financieras diferentes, la combinación de todos estos elementos hace que los esfuerzos de comparación sean muy desafiantes. Es difícil decir qué algoritmo central (PG o GE) ofrece un mejor rendimiento, es más sencillo, flexible, robusto, si agregar las extensiones citadas tiene sentido, o incluso si su impacto es similar o asimétrico, dependiendo de la elección que se haga. Se tiene la intención, en primer lugar, de llenar este vacío en la literatura presentando un ejercicio exhaustivo de evaluación comparativa.

En este trabajo experimental, se ha comparado la rentabilidad y la complejidad alcanzadas en las reglas de inversión, generadas mediante el empleo de la PG, más específicamente la STGP, y las GE. Se han utilizado, como entrada, datos diarios de sesiones bursátiles de doce años correspondientes al índice S&P 500, divididos en tres períodos consecutivos de cuatro años. Estas muestras se han diferenciado en el grado de dificultad que han ofrecido para encontrar reglas de inversión, con el potencial de vencer una estrategia de referencia basada en B&H. Para cada una de los algoritmos, se han probado cuatro configuraciones diferentes: el algoritmo central de CE utilizado; el algoritmo básico ampliado con control de parsimonia paramétrico lexicográfica; el algoritmo básico ampliado con técnicas de ADF y, finalmente, el algoritmo básico con ambas extensiones anteriores. Además de eso, se ha probado la sensibilidad de los resultados con poblaciones más grandes, y con un mayor número de indicadores técnicos formando parte del conjunto de funciones terminales. Para cumplir con este último requisito, se han agregado puntos de pivote y promedios móviles a la función inicial, y al conjunto de terminales sugeridos por Lohpetch y Corne [121].

El rendimiento de los algoritmos ha dependido del período de tiempo seleccionado para el entrenamiento y prueba, pero han sido consistentes en el sentido de que las dificultades para encontrar reglas de inversión rentables, afectó a ambos algoritmos centrales de la misma manera. Si consideramos el resultado de los 192.000 experimentos realizados, ambos algoritmos ofrecieron retornos promedio que han sido muy similares. Dicho esto, las GE funcionaron mejor, y también resultaron ser más fiables en el sentido de que la proporción de estrategias que obtuvieron rendimientos iguales o mayores que la estrategia de referencia B&H fue mayor. La estructura de los modelos de inversión obtenidos también fue diferente. Las GE



construyeron individuos más simples, tanto en términos de profundidad promedio del árbol sintáctico que los representa, como de número de nodos del mismo. Lo que se traduce en reglas de inversión más sencillas de interpretar.

Con respecto a los beneficios de extender los algoritmos básicos con el control de parsimonia paramétrico lexicográfica, las configuraciones que usaron este mecanismo de control de complejidad ofrecieron mejores rendimientos promedio, que las que no hicieron uso del mismo. A pesar del hecho de que la afirmación anterior se aplica a ambos métodos de las GE y la PG, estos últimos tendían a construir árboles mayores y se beneficiaron más de ello. El impacto de las ADF fue limitado en términos absolutos y, aunque mejoró el rendimiento promedio de las estrategias basadas en la PG, resultó tener un impacto negativo en las GE. Si fijamos la atención en la fiabilidad, ambos mecanismos proporcionaron valor añadido a las soluciones, ya que aumentaron la probabilidad de obtener resultados que fuesen al menos tan buenos como B&H. La contribución promedio de las extensiones, que se obtuvieron en los experimentos fue la misma, pero la distribución de estas fue diferente. El control de parsimonia proporciona los mismos beneficios, tanto a la PG como a las GE, pero los primeros absorbieron la mayor parte de la contribución positiva del uso de las ADF.

Los resultados experimentales, de las pruebas de sensibilidad, mostraron que la extensión de la población inicial hizo disminuir la solidez de las reglas basadas en la PG, y también resultó en una disminución media en la proporción de reglas basadas en la GE que ofrecieron un retorno igual o superior a B&H. A pesar de esto, el retorno promedio permaneció estable para la PG, mientras que en las GE se vio ligeramente afectado de forma negativa. Con respecto a los cambios en el conjunto de terminales, la adición de puntos de pivote y nuevos promedios móviles mejoró el rendimiento promedio de la PG, a costa de una pequeña disminución en las GE. Sin embargo, este cambio ayudó a ambos algoritmos básicos en términos de robustez de las estrategias de inversión, aunque no en la misma cantidad. La PG mostró capacidades mucho mejores para explotar la nueva información. Esto aumenta la importancia de las distribuciones y la probabilidad de obtener estrategias especialmente malas.

Por último, mencionar, que el uso de las gramáticas resultó más flexible a la hora de definir los posibles árboles sintácticos de los individuos y el control de tipos, frente a la programación genética con control fuerte de tipos.

Se confía que estos resultados ayudarán a los futuros investigadores y profesionales a tomar una decisión más informada, con respecto a algunos de los componentes clave que se utilizarán en el desarrollo de sus sistemas de inversión, basados en el uso de técnicas de PG y GE. También se espera, que otros autores, sigan el ejemplo y repliquen el estudio de otros activos y períodos para acumular evidencia más valiosa en diferentes condiciones de mercado. Eso, junto con un nuevo análisis sobre la sensibilidad a otros aspectos como operadores específicos, los límites en la complejidad o la parametrización de nuevas funciones técnicas darían como resultado un progreso significativo hacia la finalización de esta imagen tan compleja.

## Capítulo 7

# Enfoque de conjunto de modelos en inversión algorítmica

### 7.1 Introducción

El mercado de valores es altamente dinámico y está sujeto a cambios estructurales importantes. La utilización de un conjunto de reglas, aprendida sobre un cierto período, no garantiza su eficacia en períodos posteriores, y menos aún cuando utilizamos esa misma regla en períodos cada vez más alejados de los utilizados en el entrenamiento. Se hace necesario, a nuestro parecer, el empleo de métodos capaces de detectar y analizar dichos cambios estructurales, para producir predicciones más apropiadas. Es este segundo planteamiento, se propone una aproximación dinámica al problema aprovechando las propiedades de los sistemas evolutivos, como la de producir no sólo una solución, sino un conjunto de soluciones alternativas, o la de ser capaces de generar nuevas soluciones, ligeramente modificadas, a medida que el contexto del problema varía. Por lo tanto, en este segundo enfoque no se utiliza un sólo modelo de inversión aplicado a varias series temporales, con el objetivo de mejorar la precisión de la señal de compra/venta, sino que, por contra, se entrena a un conjunto de modelos, cada uno de ellos sobre un marco temporal diferente para aumentar la solidez de un sistema de inversión basado en AG. Se busca así, de este modo, una alternativa que se espera combine la posibilidad de cambiar el modelo, como una reacción a los cambios en el mecanismo de generación de precios, pero sin la limitación de que este cambio sea constante. Se comparará para ello el rendimiento de cuatro métodos de decisión, dos de los

cuales combinan las señales del conjunto, para generar la salida del sistema de inversión final: uno utiliza un sistema de inercia para controlar el cambio de la señal de compra de la salida del conjunto; otro se basa en un sistema de votación por mayoría simple; el tercero proviene del modelo más cercano del conjunto al día de la aplicación; y el último método utiliza el modelo más cercano del conjunto para un período completo de prueba dado.

La solución que se propone hace uso de la técnica de decisión de conjuntos (ensambles). Un conjunto, en este contexto, puede compararse con un comité de decisión colegiada, en el que se combinan los votos de varios “jueces” para llegar a una decisión final [222]. La idea, detrás de este método, es aprovechar el buen comportamiento local de cada uno de los jueces, con el fin de aumentar la precisión y la fiabilidad en el entorno de un escenario global [165].<sup>1</sup>

Las soluciones de conjuntos se emplean ampliamente en inteligencia artificial, especialmente en redes neuronales (Hansen y Salamon (1990) [86]; Perrone y Cooper (1992) [163]; Opitz y Shavlik (1996) [156]). En estos casos, se emplean varios clasificadores, generalmente redes neuronales con diferentes topologías y/o parámetros, para clasificar el mismo patrón de entrada, y sus votos se combinan usando alguna regla específica como mayoría, media aritmética, promedio ponderado, etc. No obstante, existen otros trabajos relacionados con la PG, como el de Grosan et al. (2006) [83], que hacen uso de la técnica de conjuntos en el contexto de obtención de modelos de inversión en entornos financieros de bolsa.

La precisión de la predicción es un tema crucial en la toma de decisiones de los sistemas de inversión financiera. En este sentido es posible citar otros trabajos, que proponen diferentes clasificadores basados en conjuntos, para predecir situaciones especiales en bolsa, como es el caso de las crisis [106], o para detectar manipulaciones en los precios de las acciones mediante actividades de compra y venta. En este último caso, los clasificadores deben analizar los cambios en los datos del mercado de valores y detectar estas situaciones, que afectan negativamente a los sistemas de predicción [106].

---

<sup>1</sup> En estadística y aprendizaje automático, los métodos de conjunto utilizan múltiples algoritmos de aprendizaje, para obtener un mejor rendimiento predictivo que el que podría obtenerse de cualquiera de los algoritmos de aprendizaje constituyentes por separado. [155,169,177] A diferencia de un conjunto estadístico, que generalmente es infinito, un conjunto de aprendizaje automático consiste sólo en un conjunto finito de modelos alternativos, pero que típicamente permite que exista una estructura mucho más flexible al combinar esas alternativas.

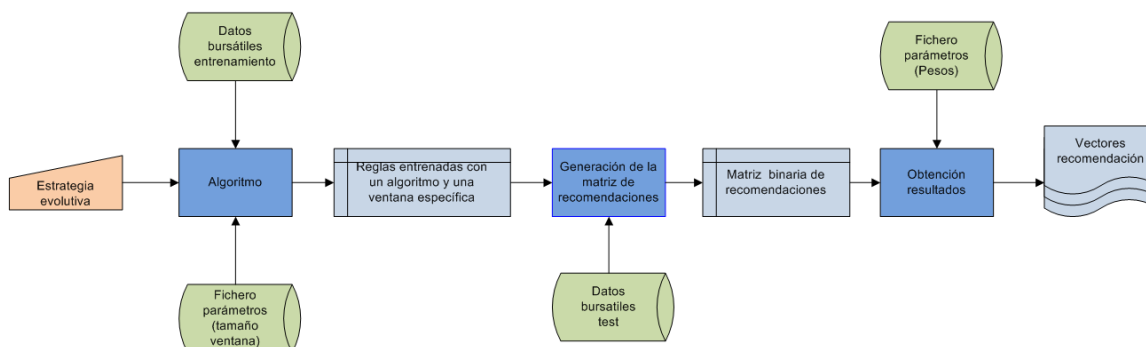
En este trabajo, el comité de decisión está formado por los diferentes modelos obtenidos mediante una ventana deslizante de entrenamiento y que componen la aproximación final proporcionada por el algoritmo de GE utilizado. La decisión final se toma en base a dos criterios diferentes: mayoría simple o mayoría cualificada, es decir, se ejecuta la operación (comprar, vender o mantenerse) a partir de la decisión que cuenta con más votos, o cuando el peso combinado de los votos es mayor.

## 7.2 Descripción del sistema

La combinación de técnicas de deslizamiento de ventanas y sistema de conjuntos desarrollada sigue el flujo de trabajo mostrado de la figura 7.1. En primer lugar se selecciona la estrategia evolutiva que se quiere utilizar, que en este caso se basará en las GE, apoyadas en el uso de control de parsimonia. Al GE, anteriormente definido, se le alimenta por un lado mediante el fichero de parámetros, que define el comportamiento del GE, y en el que se definen otros elementos como el tamaño de la ventana, y por otro lado, con el conjunto de datos de entrenamiento. El resultado del paso anterior va a generar un conjunto de reglas, cada una de ellas entrenada con el algoritmo genético elegido y una ventana de entrenamiento específica. A partir del conjunto de reglas entrenadas, y el conjunto de datos bursátiles de prueba, se procede a la generación de la matriz de recomendaciones, compuesta por una serie de "1" y "0", donde "1" significa entrar o permanecer en el mercado, y "0" salir a efectivo o permanecer en él. La matriz de recomendaciones constituye la entrada al sistema de votación. Un aspecto importante del sistema de votación es que puede ser múltiple, en el sentido que es posible alimentar la matriz a varios sistemas de votación diferentes, que puedan ser combinados mediante una técnica de conjuntos diferentes. La obtención de resultados representa la aplicación de uno o varios sistemas de votación combinados, y su salida será un vector de recomendaciones que señala, para cada una de las sesiones del período de prueba, un orden de compra o venta.

En los sistemas tradicionales, se genera una sola regla de inversión, que se utiliza en todo el período de prueba, ese sería el caso de una estrategia *Estática*, que se usará como valor de referencia. En el sistema de ventanas, por otra parte, se va obteniendo una regla por cada una de las ventanas utilizada, lo que implica que

Figura 7.1: Flujo de trabajo del sistema de conjuntos.



vayamos incrementando el número de reglas obtenidas a medida que la ventana se va desplazando. En el sistema *Naif* se utilizan múltiples reglas, pero no utiliza conjuntos, sino que cada uno de los días del período de prueba emplea una única regla, que se corresponde con la última de esas ventanas. Esta ventana finaliza el día anterior a su momento de aplicación por el sistema *Naif*. Los dos métodos de conjunto que se han utilizado, emplean un sistema de votación para obtener la decisión a partir de un número determinado de reglas, que como en el caso de *Naif* son las más cercanas al punto de aplicación. Una de las técnicas, *Mayoría*, se vale de la votación por mayoría simple, para determinar el sentido de la recomendación. La votación, en el sistema de *Mayoría*, establece el umbral en el 50 % de las recomendaciones, mientras que el sistema de *Inercia* establece el umbral en un valor superior a ese punto medio, con el objeto de ofrecer una resistencia al cambio de tendencia.

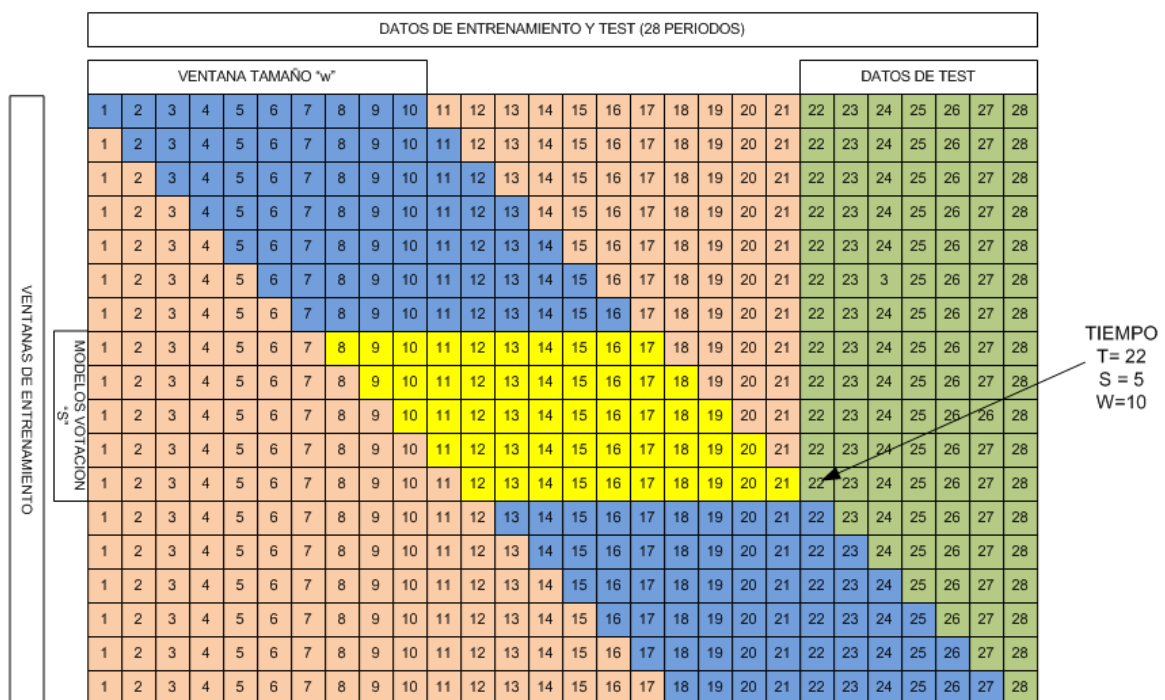
Entrando más en detalle, el planteamiento que se propone implica el uso de un conjunto de reglas de inversión, obtenidas mediante una ventana deslizante, sobre el período de entrenamiento. Se parte de una ventana de tamaño fijo,  $w$ , y se evoluciona una regla de inversión mediante GE. La regla, así obtenida, se usará en períodos subsiguientes para generar recomendaciones de inversión. Si ahora se mueve la ventana deslizante, el paso de tiempo correspondiente a un período de entrenamiento<sup>2</sup>, es posible generar un nuevo modelo basado en una nueva muestra de entrenamiento. El modelo, así generado, se superpone con el anterior para todos los elementos de entrenamiento menos el último, que será nuevo (el primer

<sup>2</sup> Podrían ser días, semanas, o meses, pero en este caso un período se corresponde con una sesión bursátil, o dicho de otro modo un día de negociación.

elemento de la ventana inicial se descarta). Una vez más, ese nuevo modelo se utilizará para generar predicciones futuras. Si sólo se considera el modelo más reciente, para emitir una recomendación en el próximo período, y se repite sucesivamente el proceso a lo largo del tiempo, se obtiene una estrategia que se denominará *Naif*.

Si se analiza el procedimiento mencionado, aún en más detalle, figura 7.2, se puede ver que en un instante de tiempo  $t$ , en el que se desea la predicción, se puede crear un conjunto de modelos que combine los  $s$  modelos generados utilizando los  $s$  períodos de entrenamiento más recientes. Esto significa que el primer modelo del conjunto se habría obtenido utilizando la parte de la serie de datos comprendida entre los instantes de tiempo  $t - s - w + 1$  y  $t - s$ , mientras que el último modelo se basaría en el período de datos que abarca el lapso temporal  $t - w$  y  $t - 1$ .

Figura 7.2: Sistema de ventanas en la generación de reglas.



(a) En sepia el conjunto de entrenamiento, en verde el conjunto de prueba, en azul las ventanas de entrenamiento utilizadas, y en amarillo las ventanas utilizadas en el conjunto para el tiempo  $t$ .

La recomendación de inversión, del conjunto mencionado, en tiempo  $t$ , requeriría combinar las recomendaciones individuales de  $s$  para ese momento. La solución más simple, para lograrlo, sería emitir un voto mayoritario entre las  $s$

recomendaciones. En caso de que la mayoría de los modelos recomendasen "1" (comprar), el conjunto de modelos aconsejaría estar en el mercado, de lo contrario, si la recomendación dominante fuese "0" (vender), el conjunto de modelos sugeriría mantener una posición de efectivo, o salir fuera del mercado. La adaptación de la idea al trabajo, a lo largo del tiempo, requeriría un conjunto de modelos diferente por cada período de tiempo. La diferencia entre uno cualquiera y su predecesor sería la adición de una nueva regla basada en los datos más recientes, y la eliminación de la más antigua para mantener la  $s$  constante y actualizada. Este sistema de inversión se ha etiquetado como *Mayoría*.

Si bien las estrategias *Naif* y *Mayoría* ofrecen la posibilidad de ser adaptativas, tienen una potencial deficiencia en común. Las reglas de inversión individuales, basadas en las GE, están optimizadas para operar en períodos de tiempo relativamente largos. Dado que la función de supervivencia de los individuos incluye los costes de los corretajes comerciales de las operaciones en su cálculo, uno de los elementos motrices del proceso de evolución es controlar el número de transacciones. Una vez que se combinan varias estrategias diferentes, este mecanismo de control implícito puede no ser tan efectivo y se corre el riesgo de realizar más operaciones de lo necesario. Esto podría dar lugar, a que la rentabilidad obtenida se vea gravemente erosionada por las comisiones.

La solución que se sugiere es proporcionar al conjunto *Mayoría* descrito un cierto grado de inercia, que sesgue las recomendaciones para intentar mantener la misma posición. Inicialmente, la primera recomendación del conjunto de modelos se obtendría por mayoría simple. A partir de ahí, sólo si existe un fuerte consenso entre las reglas del conjunto, con respecto a la necesidad de cambiar la posición actual, y, por lo tanto, entrar o salir del mercado, el conjunto de modelos lo recomendaría. La fuerza de este consenso se controlará mediante un parámetro al que en este trabajo se denomina como factor de inercia,  $i$ . Exclusivamente en el caso de que el número de modelos que recomiendan un cambio en la posición actual es mayor o igual que el umbral mencionado, el sistema lo hará. Si ese no fuera el caso, la recomendación sería mantener la posición actual, sea esta permanecer dentro o fuera del mercado.

El proceso se ilustra en la figura 7.3. Allí se presenta un ejemplo, en el que se usa un conjunto de modelos de tamaño  $s = 5$  (número de filas en amarillo), que han sido optimizados mediante muestras de entrenamiento definidas por una



Figura 7.3: Ilustración del comportamiento del conjunto de modelos.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
1	0	0	1	1	1	1	1	0	0	0	0	0	1	1	1	0	0	0	0	1	
2	0	0	1	1	1	0	1	1	1	0	0	0	0	0	0	1	1	1	0	0	
3	0	0	1	1	1	0	1	1	1	0	0	0	0	0	0	1	1	0	0	0	
4	0	1	1	1	1	0	1	1	1	1	0	0	0	0	0	1	0	0	1	1	
5	0	0	1	1	1	0	1	1	1	1	0	0	0	0	0	0	0	1	1	1	
6	0	1	1	1	1	0	1	1	0	1	1	0	1	0	0	0	0	1	1	0	
7	0	0	1	1	1	1	1	0	0	1	1	0	1	0	0	0	1	0	0	0	
8	0	1	1	1	1	0	0	0	0	1	1	1	1	1	0	0	1	0	1	1	
9	0	0	1	1	1	1	0	0	0	1	1	1	0	0	1	1	1	0	1	1	
10	0	1	1	1	1	0	0	0	0	0	1	1	1	0	1	1	1	0	0	0	
T=1, S = 5, W=10, R = Vector de recomendación															R	1					
															T	1	2	3	4	5	6

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
1	0	0	1	1	1	1	1	0	0	0	0	0	1	1	1	1	0	0	0	1	
2	0	0	1	1	1	0	1	1	1	0	0	0	0	0	1	1	1	0	0	0	
3	0	0	1	1	1	0	1	1	1	0	0	0	0	0	0	1	1	0	0	0	
4	0	1	1	1	1	0	1	1	1	1	0	0	0	0	1	0	0	1	1	0	
5	0	0	1	1	1	0	1	1	1	1	0	0	0	0	0	0	0	1	1	1	
6	0	1	1	1	1	0	1	1	0	1	1	0	1	0	0	0	0	1	1	0	
7	0	0	1	1	1	1	1	0	0	1	1	0	1	0	0	0	1	0	0	0	
8	0	1	1	1	1	0	0	0	0	1	1	1	1	1	0	0	1	0	1	1	
9	0	0	1	1	1	1	0	0	0	1	1	1	0	0	1	1	1	0	1	1	
10	0	1	1	1	1	0	0	0	0	0	1	1	1	0	1	1	1	1	0	0	
T=2, S = 5, W=10, R = Vector de recomendación															R	1	1				
															T	1	2	3	4	5	6

(a) T1. Mayoría: 1. Apoyo: 4/5. Actual: N/A Recomendación: 1 (b) T2. Mayoría: 0. Apoyo: 3/5. Actual: 1. Recomendación: 1

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
1	0	0	1	1	1	1	1	0	0	0	0	0	1	1	1	1	0	0	0	1	
2	0	0	1	1	1	0	1	1	1	0	0	0	0	0	1	1	1	0	0	0	
3	0	0	1	1	1	0	1	1	1	0	0	0	0	0	1	1	0	0	0	0	
4	0	1	1	1	1	0	1	1	1	1	0	0	0	0	1	0	0	1	1	0	
5	0	0	1	1	1	0	1	1	1	1	0	0	0	0	0	0	0	1	1	1	
6	0	1	1	1	1	0	1	1	0	1	1	0	0	0	0	0	0	1	1	0	
7	0	0	1	1	1	1	1	0	0	1	1	0	0	0	0	0	1	0	0	0	
8	0	1	1	1	1	0	0	0	0	1	1	1	1	1	0	0	1	0	1	1	
9	0	0	1	1	1	1	0	0	0	1	1	1	0	0	1	1	1	0	1	1	
10	0	1	1	1	1	0	0	0	0	0	1	1	1	0	1	1	1	0	0	0	
T=3, S = 5, W=10, R = Vector de recomendación															R	1	1	0			
															T	1	2	3	4	5	6

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
1	0	0	1	1	1	1	1	0	0	0	0	0	1	1	1	1	0	0	0	1	
2	0	0	1	1	1	0	1	1	1	0	0	0	0	0	1	1	1	0	0	0	
3	0	0	1	1	1	0	1	1	1	0	0	0	0	0	1	1	0	0	0	0	
4	0	1	1	1	1	0	1	1	1	1	0	0	0	0	1	0	0	1	1	0	
5	0	0	1	1	1	0	1	1	1	1	0	0	0	0	0	0	0	0	1	1	
6	0	1	1	1	1	0	1	1	0	1	1	0	1	0	0	0	0	1	1	0	
7	0	0	1	1	1	1	1	0	0	1	1	0	1	0	0	0	0	1	0	0	
8	0	1	1	1	1	0	0	0	0	1	1	1	1	1	0	0	1	0	1	1	
9	0	0	1	1	1	1	0	0	0	1	1	1	0	0	1	1	1	0	1	1	
10	0	1	1	1	1	0	0	0	0	0	1	1	1	0	1	1	1	0	0	0	
T=4, S = 5, W=10, R = Vector de recomendación															R	1	1	0	0		
															T	1	2	3	4	5	6

(c) T3. Mayoría: 0. Apoyo: 4/5. Actual: 1. Recomendación: 0 (d) T4. Mayoría: 1. Apoyo: 3/5. Actual: 0. Recomendación: 0

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
1	0	0	1	1	1	1	1	0	0	0	0	0	1	1	1	1	0	0	0	1	
2	0	0	1	1	1	0	1	1	1	0	0	0	0	0	1	1	1	0	0	0	
3	0	0	1	1	1	0	1	1	1	0	0	0	0	0	1	1	0	0	0	0	
4	0	1	1	1	1	0	1	1	1	1	0	0	0	0	1	0	0	1	1	0	
5	0	0	1	1	1	0	1	1	1	1	0	0	0	0	0	0	0	1	1	1	
6	0	1	1	1	1	0	1	1	0	1	1	0	0	0	0	0	0	0	1	1	
7	0	0	1	1	1	1	1	0	0	1	1	0	0	0	0	0	1	0	0	0	
8	0	1	1	1	1	0	0	0	0	1	1	1	1	1	0	0	1	0	1	1	
9	0	0	1	1	1	1	0	0	0	1	1	1	0	0	1	1	1	0	1	1	
10	0	1	1	1	1	0	0	0	0	0	1	1	1	0	1	1	1	0	0	0	
T=5, S = 5, W=10, R = Vector de recomendación															R	1	1	0	0	1	
															T	1	2	3	4	5	6

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
1	0	0	1	1	1	1	1	0	0	0	0	0	1	1	1	1	0	0	0	1	
2	0	0	1	1	1	0	1	1	1	0	0	0	0	0	1	1	1	0	0	0	
3	0	0	1	1	1	0	1	1	1	0	0	0	0	0	1	1	0	0	0	0	
4	0	1	1	1	1	0	1	1	1	1	0	0	0	0	1	0	0	1	1	0	
5	0	0	1	1	1	0	1	1	1	1	0	0	0	0	0	0	0	0	1	1	
6	0	1	1	1	1	0	1	1	0	1	1	0	1	0	0	0	0	0	1	1	
7	0	0	1	1	1	1	1	0	0	1	1	0	1	0	0	0	0	1	0	0	
8	0	1	1	1	1	0	0	0	0	1	1	1	1	1	0	0	1	0	1	1	
9	0	0	1	1	1	1	0	0	0	1	1	1	0	0	1	1	1	0	1	1	
10	0	1	1	1	1	0	0	0	0	0	1	1	1	0	1	1	1	0	0	0	
T=6, S = 5, W=10, R = Vector de recomendación															R	1	1	0	0	1	
															T	1	2	3	4	5	6

(e) T5. Mayoría: 1. Apoyo: 4/5. Actual: 0. Recomendación: 1 (f) T6. Mayoría: 0. Apoyo: 3/5. Actual: 1. Recomendación: 1

(g) Para los períodos del 1 al 6. Conjunto de 5 modelos con un umbral fijado en 4/5.

(h) Períodos de entrenamiento en color azul. Conjunto de modelos en amarillo. Períodos de prueba en verde. Recomendación de inversión: "1" permanecer en el mercado, "0" permanecer en efectivo.

(i) Recomendaciones parciales de los elementos del conjunto en rojo. Recomendación del conjunto, a continuación, en vector de resultados R.

ventana deslizante de tamaño  $w = 10$  (número de columnas de una fila en azul o amarillo). Al conjunto así entrenado, se le ha aplicado un factor de inercia fijado en  $i = 4/5$ , para generar, finalmente, el vector de recomendaciones  $R$  de las  $T = 6$  sesiones de prueba (columnas en verde). La recomendación para T1, descrita en el panel 7.3a, es el resultado de la mayoría simple. Dado que las primeras cuatro reglas más cercanas (en rojo) recomiendan estar en el mercado, y sólo la quinta recomienda una posición de efectivo (filas del 1-5), la recomendación del conjunto será mantener completamente la inversión en el mercado. Con respecto a T2, en el panel 7.3b, las cinco reglas de inversión más recientes son las numeradas del 2-6, y la mayoría de ellas recomiendan estar fuera del mercado. Sin embargo, un soporte de  $3/5$  no cumple con el criterio de umbral  $4/5$  y, por lo tanto, no es suficiente para cambiar la posición de inversión actual. Por esa razón el conjunto recomendará quedarse en el mercado. A la inversa, en T3, en 7.3c,  $4/5$  de los componentes del conjunto (reglas de inversión 3-7), recomiendan cambiar de "1" a "0". Como el soporte es mayor o igual al factor de inercia de  $4/5$ , el conjunto recomendará abandonar el mercado. La dinámica en T4-T6, representada en 7.3d - 7.3f es similar. sólo si el soporte para un cambio en la posición actual de las cinco reglas, dentro de cada conjunto, cumple con el umbral requerido, el sistema lo recomendará. El vector resultante, expuesto en el ejemplo, mediante el sistema de inercia propuesto será el "110011", mientras que el vector de voto mayoritario simple sería "100110".

## 7.3 Análisis experimental

En esta sección, se describirá la configuración experimental que incluye: el conjunto de datos de entrada de entrenamiento y prueba utilizados; el protocolo experimental empleado; la parametrización elegida; seguido de la presentación y discusión de los resultados obtenidos.

### 7.3.1 Conjunto de datos y diseño experimental

La validación experimental, del sistema descrito, se realizará evaluando su rendimiento sobre una serie de datos históricos, comparando sus resultados frente a otras estrategias comparables basadas en las GE, más específicamente: la *Estática*, la *Naif* y la de *Mayoría* descritas en la sección 7.

En lo relativo a la muestra experimental utilizada, el estudio considerará dos conjuntos de datos que cubren el lapso temporal de 13 años, desde el año 2005 al 2017. El primero conjunto de datos corresponde al índice Standard & Poor's 500, obtenido de Datastream, y el segundo el diario de retorno sin riesgo, obtenido del Banco de la Reserva Federal de Atlanta y disponible en la URL: <https://fred.stlouisfed.org/series/TB3MS>.

Como ya se mencionó en la introducción, uno de los objetivos de este trabajo es desarrollar un sistema de inversión dinámico basado en el uso de las GE para mejorar el planteamiento estático tradicional de la PG y las GE que, tras el entrenamiento, desarrollan una regla de inversión única para generar recomendaciones de inversión dirigidas a períodos de tiempo predeterminados. La principal ventaja de la alternativa que se ha introducido es que permite al sistema ser capaz de hacer frente a los cambios estructurales de los mercados automáticamente. Lo que se lleva a cabo a través de un conjunto de modelos entrenados, a partir de series de marcos de tiempo múltiples.

El proceso comenzará con la extracción de dos submuestras del conjunto de datos original. La primera comprende desde el año 2005 al 2012, y se utilizará para fines de parametrización del algoritmo de GE. A partir de ella, se establecerán los parámetros básicos para los algoritmos de las GE y, basándonos en la realización de experimentos preliminares, se elegirá la cantidad de modelos, que se incluirán en el conjunto  $s$  de votación, y el valor del factor de inercia utilizado,  $i$ . Una vez que se determinen estos valores, se ejecutarán los experimentos principales utilizando una segunda submuestra en el período que va del año 2010 al 2017. Vale la pena señalar, que la superposición parcial de las dos submuestras tiene que ver con el tamaño de la ventana deslizante para evolucionar las reglas. Se han utilizado datos diarios de tres años, por lo que los períodos de evaluación, para fines de parametrización y de evaluación comparativa no se superponen. La comparación de los cuatro métodos se basará en los retornos netos de inversión obtenidos en los años 2013, 2014, 2015, 2016 y 2017

Dado que las GE constituyen un método estocástico, los experimentos se realizarán 30 veces, tal como se expuso en el capítulo 5. La significación estadística de los resultados se probará formalmente, mediante el protocolo expuesto en dicho capítulo 5, y que por conveniencia se presenta resumido a continuación:

Inicialmente, la normalidad de los retornos se evaluará mediante la prueba

de *Kolmogorov-Smirnov*, aplicando la corrección de *Lilliefors*. En caso de que se rechace la normalidad de las observaciones se aplica la prueba de rangos de signos no paramétrica de *Wilcoxon*. De lo contrario la prueba de *Levene* comprueba la homocedasticidad de las varianzas. Si hay homocedasticidad se aplica la prueba *t* de *student*, en caso de que no la haya se aplicará la prueba *t* de *Welch*.

### 7.3.2 Parametrización

La parametrización general del algoritmo de las GE utilizado en esta tesis, está descrita, en más detalle, en el capítulo 4, reflejándose aquí un resumen de los parámetros más importantes, con las particularidades referentes a la experimentación de la solución propuesta de conjuntos de modelos en inversión algorítmica.

La parametrización del algoritmo se estableció tras la realización de algunos experimentos preliminares, a los que se añadieron, para su uso, otros valores de parámetros comúnmente encontrados en la literatura. Como tamaño de la población se seleccionaron 500 individuos; se implementaron ciclos evolutivos de 50 generaciones y se incluyó la técnica de elitismo de un único individuo; en cada generación se seleccionó la mejor estrategia obtenida y se copió sin mutar en la población del siguiente ciclo evolutivo. Los individuos fueron inicializados usando series geométricas. En la inicialización de la población se estableció un valor de complejidad inicial mínimo de cinco y una probabilidad de crecimiento de 0,85.

Para mejorar la variabilidad en la población, y evitar una convergencia prematura, tanto durante el proceso de inicialización del algoritmo como durante la operación genética de mutación, si el mismo individuo (misma carga genética) aparecía más de una vez en la población, se efectuaban hasta 100 intentos de reemplazarlo por uno nuevo.

El tamaño de la ventana de entrenamiento se estableció en tres años (753 sesiones), mientras que el de número de sesiones del período de prueba se fijó en un año (251 sesiones).

Con respecto al número de modelos, que se utilizaron en el conjunto para obtener la predicción, inicialmente se realizaron algunos experimentos de parametrización exploratoria, utilizando como referencia el año 2012. Las combinaciones, que se consideraron, incluyeron conjuntos de 3, 5, 15 y 25 modelos y factores de

inercia de 2/3, 3/4 y 4/5. Para todas las configuraciones los experimentos se repitieron 30 veces. Los mejores resultados de estas pruebas, en términos de retornos de inversión absolutos, se obtuvieron con 3 modelos y unanimidad. Cabe señalar que, para ese tamaño de conjunto de 3 modelos en particular, tanto los factores de inercia de 3/4, 4/5 y unanimidad se comportan de forma idéntica.

Con el fin de asegurarse de que estos resultados fueran razonablemente estables, a lo largo del tiempo, se extendió el análisis calculando el promedio de los rendimientos anuales, para todos los períodos que terminaban a finales del 2012, a partir de 2008 en adelante.

Estos resultados se resumen en la Tabla 7.1. Como es posible ver, independientemente de si se considera el período de uno o cinco años, la configuración elegida parece ser adecuada. Ese es también el caso para períodos de 2, 3 y 4 años.

**Tabla 7.1:** Parametrización de los tests de conjuntos.

Modelos	Factor de inercia	2008-12	2009-12	2010-12	2011-12	2012
3	2/3	-0,0356	0,0097	0,0032	0,0443	0,1734
3	3/4	-0,0128	0,0446	0,0356	0,0742	0,2043
3	4/5	-0,0128	0,0446	0,0356	0,0742	0,2043
5	2/3	-0,0307	0,0279	0,0133	0,0423	0,1955
5	3/4	-0,0307	0,0279	0,0133	0,0423	0,1955
5	4/5	-0,0359	0,0390	0,0287	0,0600	0,1667
15	2/3	-0,0574	0,0249	0,0146	0,0461	0,1523
15	3/4	-0,0764	0,0144	0,0021	0,0240	0,1487
15	4/5	-0,0729	0,0191	0,0081	0,0336	0,1483
25	2/3	-0,0642	0,0356	0,0306	0,0482	0,1265
25	3/4	-0,0739	0,0294	0,0215	0,0452	0,1260
25	4/5	-0,0891	0,0149	0,0045	0,0326	0,1403

(a) Rendimiento anual promedio en 30 experimentos, para las diferentes combinaciones del número de modelos de los que se componen los conjuntos, y el factor de inercia utilizado durante los períodos especificados.

(b) Mejores resultado mostrados en azul claro.

### 7.3.3 Resultados experimentales

Los resultados de los experimentos se presentan en la tabla de rendimientos netos 7.2, donde se incluyen los costes de transacción, y en la tabla 7.4 en donde no se incluyen. Allí se proporcionan las principales estadísticas descriptivas, de los retornos obtenidos, en las muestras de prueba de 30 experimentos durante cinco años. Además del enfoque de conjunto con inercia, que es objeto del análisis, se presenta el rendimiento obtenido en otros tres puntos de referencia: la solución estática estándar, donde se obtiene una regla de negociación única entrenada durante los tres años más cercanos al período de prueba, y que se usa posteriormente para el conjunto de un año de prueba, *Estático*; un planteamiento que usa un modelo diferente, lo más cercano posible al punto de predicción, por cada uno de los pasos de tiempo, *Naif*; y los resultados de un conjunto, que se basa en una votación por mayoría simple, para hacer las recomendaciones, *Mayoría*. Este último, considera los mismos modelos que el basado en la perspectiva que contempla la *Inercia*. La diferencia entre ambas soluciones propuestas es la forma en que el sistema combina las recomendaciones de los modelos para producir la señal de salida final.

Como se puede ver, en los resultados de la tabla de rendimientos netos 7.2, el conjunto de modelos con el componente de *Inercia* obtiene el mejor rendimiento promedio de los cuatro métodos analizados. Si se considera el rendimiento medio anual, durante los cinco años, este enfoque obtuvo un 4,77% de rendimiento en comparación con el 2,55% de la propuesta *Estática*, y el 2,09% del conjunto que hace recomendaciones basadas en la mayoría de los votos, *Mayoría*. El sistema, que presentó el rendimiento más bajo, con una pérdida neta anual promedio del 3,96%, implicaba un entrenamiento utilizando una ventana deslizante, para entrenar reglas de inversión que sólo se usaban una vez en cada paso de predicción, *Naif*. La estrategia de *Inercia*, también ofreció ventajas en términos de incertidumbre, ya que el promedio de las desviaciones anuales es aproximadamente la mitad, 0,00054, en comparación con el 0,00099 del segundo enfoque *Estático*.

Sin embargo, existe una cierta variabilidad una vez que se analizan los resultados año por año. Mientras que el sistema basado en *Inercia* domina claramente sobre el resto en tres de los cinco años analizados, la estrategia *Estática* superó al resto en los años 2015 y 2016, que también fueron los años en los que el mercado ofreció el peor rendimiento. Aunque el rango de cada solución, *Estática*, *Naif* y *Mayoría*, cambia con el tiempo, la basada en *Inercia* domina consistentemente sobre los

**Tabla 7.2:** Retorno neto de las estrategias de conjuntos de modelos.

	Estrategia	Media	Mediana	Var.	Max.	Min.
2013	Inercia	0,1423	0,1400	0,0006	0,1691	0,1298
	Estática	0,0381 **	0,0409	0,0001	0,0714	0,0051
	Naif	0,1311 **	0,1323	0,0003	0,1715	0,0909
	Mayoría	0,1367 *	0,1363	0,0002	0,1557	0,0964
2014	Inercia	0,0947	0,0938	0,0006	0,1048	0,0749
	Estática	0,0606 **	0,0827	0,0015	0,1042	0,0079
	Naif	0,0547 **	0,0526	0,0002	0,0813	0,0281
	Mayoría	0,0852 **	0,0876	0,0001	0,0969	0,0471
2015	Inercia	-0,0511	-0,0468	0,0006	-0,0214	-0,1327
	Estática	-0,0160 **	-0,0123	0,0001	-0,0123	-0,0600
	Naif	-0,1687 **	-0,1685	0,0024	-0,0574	-0,2564
	Mayoría	-0,0903 **	-0,0817	0,0017	-0,0303	-0,1738
2016	Inercia	-0,0089	-0,0083	0,0011	0,0579	-0,0731
	Estática	0,0241 **	0,0043	0,0016	0,0889	-0,0223
	Naif	-0,1455 **	-0,1384	0,0022	-0,0794	-0,2582
	Mayoría	-0,0448 **	-0,0383	0,0012	0,0075	-0,1231
2017	Inercia	0,0615	0,0621	0,0009	0,1076	0,0065
	Estática	0,0204 **	0,0092	0,0016	0,1668	0,0091
	Naif	-0,0718 **	-0,0755	0,0011	0,0295	-0,1316
	Mayoría	0,0176 **	0,0294	0,0022	0,1094	-0,0753
Promedios	Inercia	0,0477	0,0481	0,00054	0,0836	0,0011
	Estática	0,0255	0,0250	0,00099	0,0832	-0,0120
	Naif	-0,0396	-0,0419	0,00104	0,0311	-0,0985
	Mayoría	0,0209	0,0267	0,00105	0,0678	-0,0458

(a) \*\* Significación estadística vs. Inercia al 1%. \* Significación estadística vs. Inercia al 5%.

(b) Incluye los gastos de transacción. Principales estadísticas descriptivas en 30 ejecuciones. Resultados de la prueba.

(c) En color azul claro se marcan las técnicas y períodos en los que se obtienen los mejores rendimientos promedio.

dos últimos, *Naif* y *Mayoría*.

La significación estadística, de las diferencias de retorno de medianas, frente a la estrategia de *Inercia*, se ha probado formalmente utilizando la prueba de *Wilcoxon*. Como podemos ver en 7.2, todas las diferencias, excepto una, son significativas al 1%. La única excepción es la diferencia entre el sistema de *Inercia* y el conjunto con el mecanismo de votación simple, *Mayoría*, en el año 2013. En ese caso, las diferencias son significativas al 5%.

Con el fin de obtener información adicional, sobre el mecanismo que podría explicar el rendimiento diferencial conseguido, se ha examinado el número de transacciones requeridas por las estrategias. Los datos proporcionados en la Tabla 7.3, ponen de manifiesto las discrepancias a este respecto entre los cuatro planteamientos utilizados.

Los dos sistemas que resultan en un menor número de transacciones son el de *Inercia* y el *Estático*, que de promedio realizan 7,36 y 4,72 transacciones. Dependiendo del período considerado uno tiene un rendimiento más alto que el otro. Sin embargo, en el otro lado del espectro, la estrategia adaptable *Naif* solicita más órdenes de compra y venta, seguida de *Mayoría*, 44,6 de *Naif* y 18,29 de *Mayoría*. Existe una clara relación inversa entre esta cifra y la rentabilidad, y, como se puede ver en los resultados, los costes de transacción socavan gravemente el rendimiento de esta solución de conjunto de modelos.

Curiosamente, el enfoque *Estático* dio como resultado un buen número de estrategias que no proporcionaron ninguna señal de compra en el año 2017 y, por lo tanto, se mantuvieron fuera del mercado durante todo el período, obteniendo así la rentabilidad libre de riesgos.

Otra evidencia, que respalda la importancia que tienen los costes de transacción en los resultados, sería el hecho de que la estrategia de *Inercia* venció a la aproximación *Estática* en los dos períodos en los que requirió menos transacciones. Sin embargo, todo esto es más evidente una vez que se considera el rendimiento bruto.



**Tabla 7.3:** Número de transacciones en estrategias de conjunto.

	Estrategia	Media	Mediana	Var.	Max	Min.
2013	Inercia	4,07	4	0,13	6	4
	Estática	7,20	4	48,17	32	4
	Naif	14,00	14	16,00	20	4
	Mayoría	5,80	6	3,96	10	4
2014	Inercia	4,93	5	2,96	8	2
	Estática	7,07	4	25,31	14	2
	Naif	14,73	14	11,72	22	8
	Mayoría	8,00	8	1,66	10	6
2015	Inercia	8,80	8	3,20	12	4
	Estática	2,40	2	1,21	6	2
	Naif	42,40	42	24,94	56	36
	Mayoría	17,00	18	10,69	26	10
2016	Inercia	11,00	12	3,79	14	8
	Estática	5,53	4	13,43	18	2
	Naif	72,47	72	58,40	92	62
	Mayoría	29,33	28	19,95	40	18
2017	Inercia	8,00	8	5,24	12	4
	Estática	1,40	0	13,28	12	0
	Naif	79,40	81	102,39	100	52
	Mayoría	31,33	32	35,40	44	20
Promedio	Inercia	7,36	7,4	3,06	10,4	4,4
	Estática	4,72	2,8	20,28	16,4	2,0
	Naif	44,60	44,6	42,69	42,0	48,40
	Mayoría	18,29	18,4	17,50	26,0	11,6

(a) Principales estadísticas descriptivas en 30 ejecuciones. Resultados de la prueba.

La tabla 7.4 representa los rendimientos brutos, es decir, las cantidades mostradas en la tabla 7.2 a las que se suman los costes derivados de transacción. Si se calculan los rendimientos promedio, durante los cinco años, se puede apreciar que la estrategia *Naif*, con un 7,21 %, ofrece los mejores resultados. Le siguen la solución basada en la *Mayoría*, con un 6,66 % y la de *Inercia*, con un 6,61 %. El enfoque tradicional, representado por la estrategia *Estática*, que ofreció el segundo mejor rendimiento promedio en términos netos, proporcionó, sin embargo, el peor rendimiento en términos brutos, con un 3,73 %.

Si se analizan los datos año por año, los resultados son variados, en primer lugar hay una mejora importante en la estrategia *Naif*, que es la mejor en 2013, y sobre todo en 2017, empatando con la estrategia *Estática* en 2016. Por otra parte, la estrategia de *Inercia* que fue la mejor cuando se contemplaban los costes de transacción, no ha sido capaz de vencer en ninguno de los períodos. Curiosamente la estrategia *Estática* lo consigue en 2015, empatando con *Naif* en 2016, mientras que la basada en *Mayoría*, que no lo consigue en ningún período, es sin embargo la que se comporta mejor en segundo lugar, en términos de promedios.

Una vez más, la aproximación representada por *Inercia* parece ofrecer los resultados más confiables. El promedio de las variaciones anuales de rendimiento de un 0,00044, es aproximadamente la mitad del segundo enfoque más estable, *Mayoría* con un 0,00087, y el promedio más alto de rendimiento mínimo de un 3,75 % frente al 3,24 % del segundo mejor resultado aportado por *Naif*.

Esta evidencia experimental brinda un fuerte apoyo a la importancia de usar enfoques adaptativos, ya que muestran un gran potencial frente a los *estáticos* tradicionales y controlan los costes de transacción, que son claramente un factor de especial relevancia. La estrategia *Inercia* presentada ofrece un buen compromiso entre dos objetivos contradictorios, ya que ofrece flexibilidad para adaptarse al cambio estructural, aunque menos que *Naif*, y limita el número de transacciones, aunque menos que la *Estática*. Al examinar el efecto de los costos de transacción, se puede ver que estos son proporcionales a las estrategias dinámicas aplicadas a las reglas de inversión de la cartera, que en este trabajo está compuesta por los títulos del índice S&P 500, para un sistema de inversión que intente maximizar el rendimiento esperado de esta. Una política de inversión óptima ha de tener en cuenta este factor, y describirlo en términos de las regiones sin transacción, donde la política óptima, a seguir, es abstenerse de negociar si el estado de la cartera se

**Tabla 7.4:** Retorno bruto de las estrategias de conjuntos.

	Estrategia	Media	Mediana	Var.	Max.	Min.
2013	Inercia	0,1525	0,1500	0,0001	0,1791	0,1398
	Estática	0,0561 **	0,0509	0,0004	0,1215	0,0193
	Naif	0,1661 **	0,1654	0,0003	0,2215	0,1409
	Mayoría	0,1512	0,1519	0,0001	0,1676	0,1214
2014	Inercia	0,1070	0,1067	0,0001	0,1248	0,0899
	Estática	0,0783	0,0927	0,0007	0,1092	0,0429
	Naif	0,0915 **	0,0890	0,0001	0,1165	0,0716
	Mayoría	0,1052	0,1074	0,0001	0,1184	0,0671
2015	Inercia	-0,0291	-0,0223	0,0001	-0,0014	-0,1077
	Estática	-0,0100 **	-0,0073	0,0001	-0,0073	-0,0451
	Naif	-0,0627 **	-0,0627	0,0024	0,0576	-0,1379
	Mayoría	-0,0478 *	-0,0415	0,0014	0,0084	-0,1288
2016	Inercia	0,0186	0,0217	0,0010	0,0779	0,0381
	Estática	0,0379	0,0293	0,0015	0,0952	0,0073
	Naif	0,0357	0,0409	0,0018	0,1346	0,0282
	Mayoría	0,0285	0,0355	0,0010	0,0775	0,0481
2017	Inercia	0,0815	0,0818	0,0009	0,1260	0,0277
	Estática	0,0239 **	0,0092	0,0017	0,1718	0,0092
	Naif	0,1296 **	-0,0708	0,0013	0,1968	0,0591
	Mayoría	0,0959	0,0294	0,0014	0,1694	0,0323
Promedio	Inercia	0,0661	0,0675	0,0004	0,1012	0,0375
	Estática	0,0373	0,0350	0,0009	0,0980	0,0067
	Naif	0,0721	0,0723	0,0012	0,1454	0,0324
	Mayoría	0,0666	0,0703	0,0008	0,1082	0,0280

(a) \*\* Significación estadística vs. Inercia al 1%. \* Significación estadística vs. Inercia al 5%.

(b) No incluye los costes de transacción. Principales estadísticas descriptivas en 30 ejecuciones. Resultados de la prueba.

(c) En color azul claro se marcan las técnicas y períodos en los que se obtienen los mejores rendimientos promedio.

encuentra dentro de la región, y realizar transacciones al límite más cercano de la región, si el estado de la cartera se encuentran fuera de la misma [75].

Debido a que los límites de las regiones sin transacción no se pueden derivar analíticamente, los algoritmos han de desarrollar reglas eficientes y simples que tengan en cuenta esos límites.

## 7.4 Resumen y conclusiones

La generación de reglas de inversión con representación flexible, utilizando las gramáticas evolutivas en su versión estándar, implica obtener una única regla, basada en un período de entrenamiento dado, que posteriormente se utiliza para generar recomendaciones, que se utilizarán durante un determinado período de tiempo. Dada la prevalencia del cambio estructural, en las series financieras, esto plantea un problema.

En este experimento, para tratar este problema, se sugiere utilizar un conjunto de reglas de inversión, obtenidas mediante gramáticas evolutivas, que se apoyan en el uso de una ventana de entrenamiento deslizante. El sistema dispone de un componente crítico, el mecanismo de votación del conjunto, que crea una inercia que reduce los casos en los que el número de transacciones de compra/venta es excesivo.

La evaluación experimental de esta solución involucró tres alternativas con las que establecer la comparación, que se apoyan en el uso del mismo algoritmo central durante un período de cinco años. Estas incluían: la estrategia estática estándar, *Estática*; una que usa un modelo diferente por paso de tiempo, *Naif*; y los resultados de un conjunto, que se basa en la mayoría simple, para efectuar las recomendaciones, *Mayoría*.

Los resultados obtenidos, apoyan la superioridad del conjunto de modelos con el componente de inercia, en términos de retorno de inversión, seguido del enfoque estático tradicional y del conjunto de modelos que implementa el voto mayoritario simple. El sistema de peor rendimiento fue el basado en la estrategia *Naif*. La estrategia sugerida, también ofrecía ventajas en términos de disminución de la incertidumbre.

El análisis del impacto de los costes de transacción hace evidente la importancia de limitar el número de las transacciones comerciales efectuadas, pues existe una clara relación inversa entre el número de órdenes de compra y venta y la rentabilidad obtenida. La estrategia *Naif* negocia mucho más a menudo que el resto de aproximaciones y, por tanto, las comisiones erosionan su rendimiento de manera muy significativa. Hasta el punto de que, a pesar de que recoge mejor los cambios estructurales y se adapta mejor a la serie, como se aprecia en los resultados que no tienen en cuenta los costes de transacción, su rendimiento es inferior.

Estos hallazgos enfatizan la importancia de mantener un equilibrio, entre la necesidad de adaptarse a los cambios estructurales y el riesgo de actualizar constantemente los modelos de recomendación de inversión, que están optimizados implícitamente para el largo plazo.

*Esta página se ha dejado en blanco intencionadamente.*

## Capítulo 8

# Selección dinámica de modelos

### 8.1 Introducción

En este tercer experimento se sugiere una alternativa, que también está basada en modelos que se entrenan utilizando un enfoque de ventana deslizante. No obstante, en esta ocasión, el enfoque adaptativo que se sugiere implica un mecanismo de selección dinámico, que conmuta entre una regla activa y una candidata, que ha sido optimizada conforme a los datos de mercado más recientes disponibles. Es probable que, en este caso, los modelos produzcan reglas que cumplen los dos objetivos perseguidos: primero que las transacciones señaladas se adapten a los cambios estructurales, y segundo que estas no se produzcan en exceso, siendo por tanto más fructíferas en términos de rentabilidad.

En la estrategia de conjuntos de *Inercia* del experimento anterior, el número de reglas, que formaban el conjunto de predicción, era variable y podía tener cualquier tamaño, que se establecía parametrizándolo antes de la ejecución del experimento. En la estrategia *dinámica*, que se propone, el conjunto de reglas utilizadas en la predicción es de tamaño fijo, cuyo valor es dos<sup>1</sup>. Las reglas más antiguas se van eliminando del sistema siguiendo un mecanismo de torneo, a medida que se va desplazando la ventana, siendo sustituidas por nuevas reglas más cercanas al punto de predicción. Este mecanismo persigue dos objetivos, por un lado, que las reglas más antiguas no sigan influyendo en las decisiones, pero al mismo tiempo que estas reglas previas establezcan un mecanismo de resistencia al cambio (histéresis), que evite una excesiva negociación.

---

<sup>1</sup> En realidad es posible ajustar el sistema para que la competición se establezca entre un número arbitrario de modelos.

El flujo de trabajo del sistema es idéntico al mostrado en la figura 7.1, cambiándose, eso sí, el método de obtención de resultados, que pasa de ser un sistema de votación a un sistema que conmuta entre dos reglas por torneo, y aplica la más rentable de las dos para establecer la predicción.

El sistema se basa en el uso de ventanas deslizantes durante el entrenamiento de los modelos. Entrando en detalle en la solución de ventanas esta se comporta de la siguiente forma: dado un tamaño de ventana  $w$  definido y constante, el proceso comienza por seleccionar  $w$  puntos de datos, en un determinado momento en el pasado, para evolucionar una regla de inversión utilizando las GE. Una vez que se obtiene una regla esta se puede utilizar para generar recomendaciones de inversión en períodos futuros. Si esto se lleva a cabo sólo una vez, entrenando las reglas con la ventana de datos de entrenamiento más cercana al período de predicción, y usamos el modelo de inversión resultante durante todo el período de prueba, tendríamos el enfoque estándar no adaptativo que se encuentra habitualmente en la literatura, y que se etiqueta como *Estático*.

En el extremo opuesto del espectro, es posible repetir el proceso para generar estrategias que sólo se usarán una vez en un único paso de tiempo. Es decir, si se moviera la ventana deslizante un paso cada vez, se obtendrían muestras de entrenamiento superpuestas que diferirían en un sólo elemento. La nueva ventana de entrenamiento agregaría el punto de datos más reciente y se eliminaría a la vez el más antiguo. Si se considera sólo la regla más reciente para proporcionar una recomendación de inversión para el próximo período, en el caso de este trabajo de una única sesión, y se itera el proceso en todo el período de prueba, se obtendría una estrategia muy adaptable que se etiqueta como *Naif*.

La solución intermedia, que se presenta en esta parte de la tesis, pretende lograr un equilibrio entre la necesidad de adaptarse a los cambios estructurales, y limitar el impacto en los resultados derivado de una negociación excesiva. El punto de partida de este enfoque sería la solución *Naif*, que evoluciona una regla de inversión para cada paso de tiempo, utilizando un sistema basado en ventanas deslizantes con un tamaño de ventana fijado en  $w$ . La solución que se propone, que como *Naif* también evoluciona una única regla de inversión, figura 8.1, parte de una ventana de tamaño fijo  $w$  que se utiliza como entrenamiento, a partir de la cual se evoluciona una regla de inversión mediante GE. La primera ventana,  $w$ , se encuentra al inicio de los datos de entrenamiento entre los puntos  $t_1$  y  $t_1 + w - 1$



(en amarillo 1-10). Dado que en ese instante de tiempo sería la única alternativa posible, esta se usaría para generar la primera recomendación de inversión y su predicción se aplicaría al punto  $t_1 + w + 1$  (en rojo 11), guardando su predicción en un vector de resultados. En  $t_2$  ya hay dos reglas disponibles, por lo que entraría en funcionamiento un sistema de torneo, entre la regla activa anterior y la que se toma como candidata, que selecciona la regla más rentable de las dos. La regla de inversión candidata se desarrollaría según la ventana de entrenamiento  $t_2$  y  $w + 1$ . En ese punto, la mejor regla activa y la nueva regla candidata se comparan, para lo cual se añade a la regla activa un punto posterior en el tiempo,  $w + 1$  (en rojo 11), y se elimina el punto más antiguo,  $t_1$  (fila 1 en amarillo columna 1). El resultado de esta comparación asignaría a la mejor regla obtenida el estatus de mejor regla activa, que se usará posteriormente para generar la recomendación de inversión requerida para el instante  $w + 2$  (en rojo 12). El proceso continua hasta alcanzar el final de los datos de prueba (última fila 18, columna en rojo 28), que se referencia como  $t_z$ . La última regla candidata sería, por tanto, aquella cuyo período iría desde  $t_z - w$  hasta  $t_z - 1$ , efectuándose la predicción en el punto  $t_z$ .<sup>2</sup>

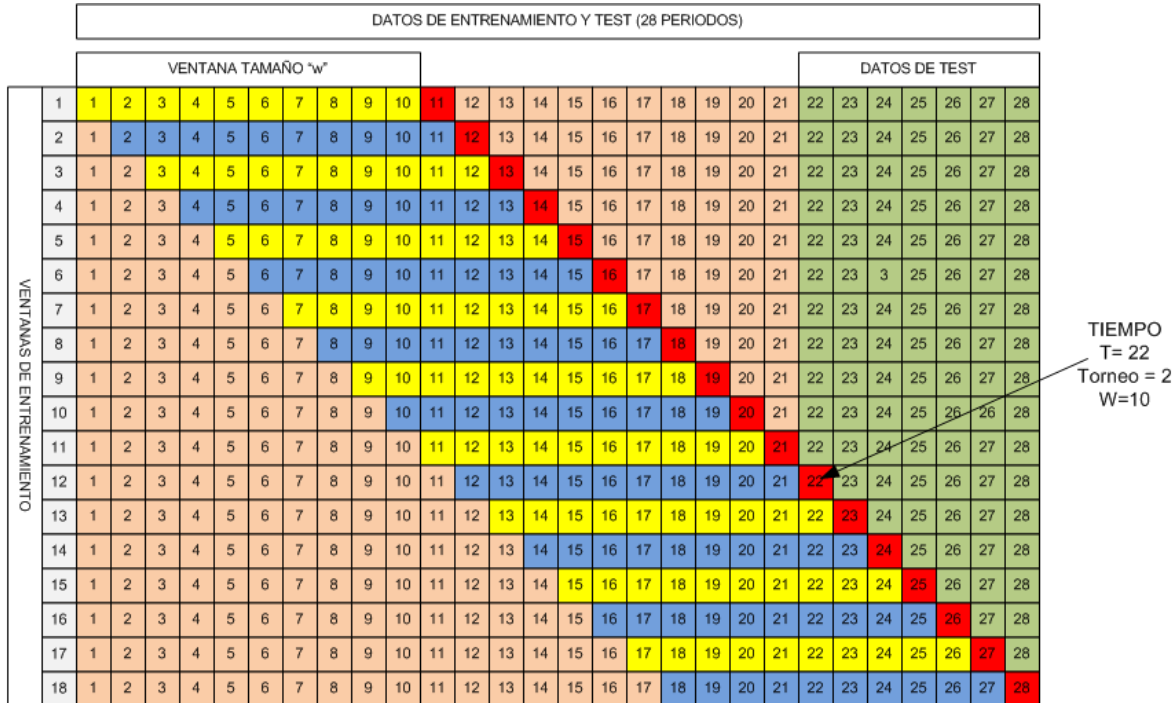
El proceso de comparar la mejor regla de inversión en curso y una nueva regla candidata, para actualizar la mejor regla actual, se repetirá con el tiempo, buscando alcanzar una estrategia de inversión *Dinámica* adaptable y eficiente en la negociación. Vale la pena señalar, que una vez que la mejor regla en curso se reemplaza por una nueva, esta se pierde, es decir es eliminada del sistema. Recalcar, que el proceso realiza una búsqueda que no considera haber alcanzado una alternativa incorrecta y, por tanto, no retrocede hasta el paso anterior para tomar la siguiente alternativa (backtracking<sup>3</sup>) sino que sólo puede avanzar en el tiempo examinando cada vez reglas candidatas más modernas.

Un elemento clave, pendiente de definir, es la forma en la que las reglas van a ser comparadas. Esto se hace sobre la base del rendimiento de la inversión realizada en un período de evaluación común, y que será siempre el período de entrenamiento para la nueva regla candidata. Esto significa que el rendimiento de la mejor regla en curso siempre incluirá una proporción de las recomendaciones de

<sup>2</sup> En este ejemplo se ha supuesto que la regla candidata siempre vence a la activa.

<sup>3</sup> La vuelta atrás o "Backtracking" [84] es una estrategia para encontrar soluciones a problemas que satisfacen restricciones, como los de satisfacibilidad lógica, en los que se van incrementando las soluciones candidatas, abandonándolas tan pronto como se determina que el candidato no puede ser completado a una solución válida. En este contexto la vuelta atrás hace referencia, a que una vez que una regla ha sido descartada ya no se vuelve a examinar.

Figura 8.1: Sistema de selección dinámica de modelos de inversión.



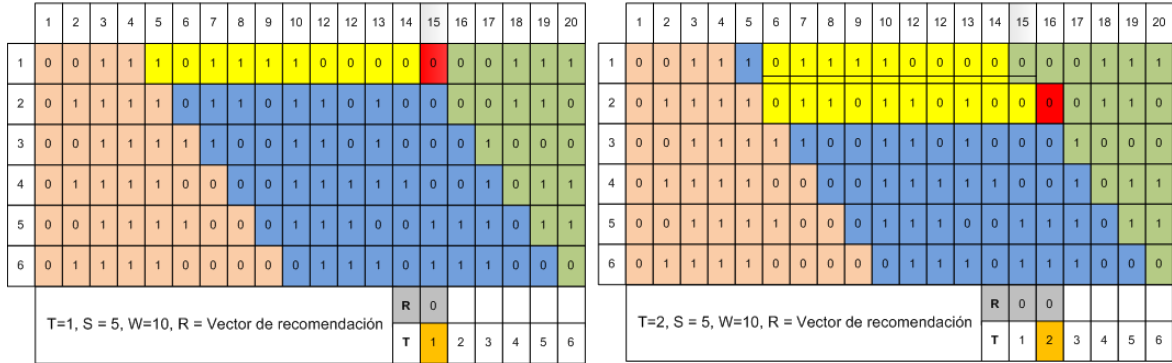
(a) En sepia el conjunto de entrenamiento, en verde el conjunto de prueba, en azul y amarillo las ventanas de entrenamiento utilizadas, y en rojo el elemento añadido para permitir la comparación de reglas sucesivas.

inversión en los datos de prueba. En el extremo inferior siempre tendremos  $w - 1$  recomendaciones en los datos de entrenamiento y una recomendación en prueba, si el mejor modelo en vigor fuera el modelo candidato en el paso de tiempo anterior, o  $s$  recomendaciones en los datos de prueba si el mejor modelo en curso ha mantenido su estatus en  $s$  o más períodos de tiempo consecutivos.<sup>4</sup>

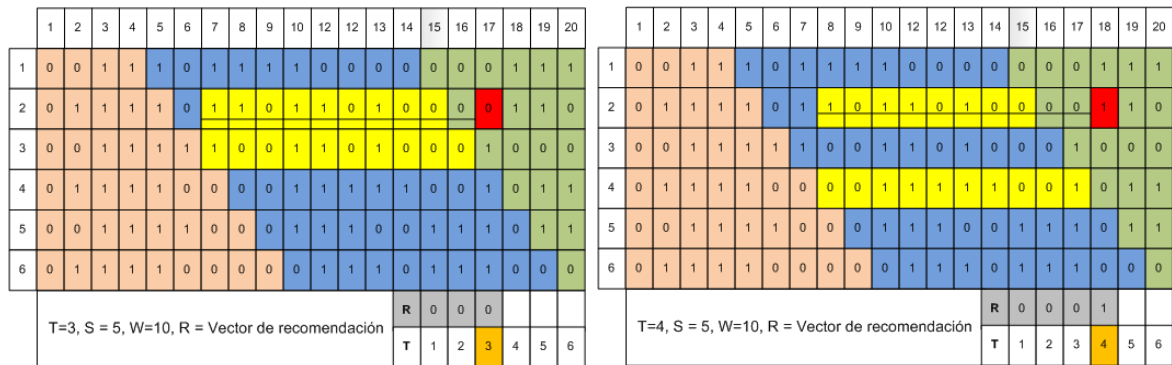
Este proceso se ilustra con un ejemplo en la figura 8.2. El panel 8.2a describe la selección de una primera regla de inversión, obtenida con las GE, en una muestra de entrenamiento de  $w = 10$  períodos,  $t_5 - t_{14}$  (en amarillo). Al ser la primera regla, esta es elegida como regla activa y generará la recomendación. Una vez que se obtiene la primera recomendación  $R$ , paso  $T1$ , "0",  $t_{15}$  (en rojo), la ventana deslizante

<sup>4</sup> Puede darse el caso, de que al contrastar el rendimiento de las dos reglas, si la mejor regla es muy antigua, al compartir el mismo período de ventana que la candidata para su contraste, las ventanas de entrenamiento sean completamente disjuntas. Esto no es un problema ya que, en realidad, tras el período de entrenamiento, el resto del conjunto de datos restante no utilizado se ha rellenado con predicciones del propio entrenamiento, es decir parten de la misma regla, pero aplicada en diferentes espacios temporales.

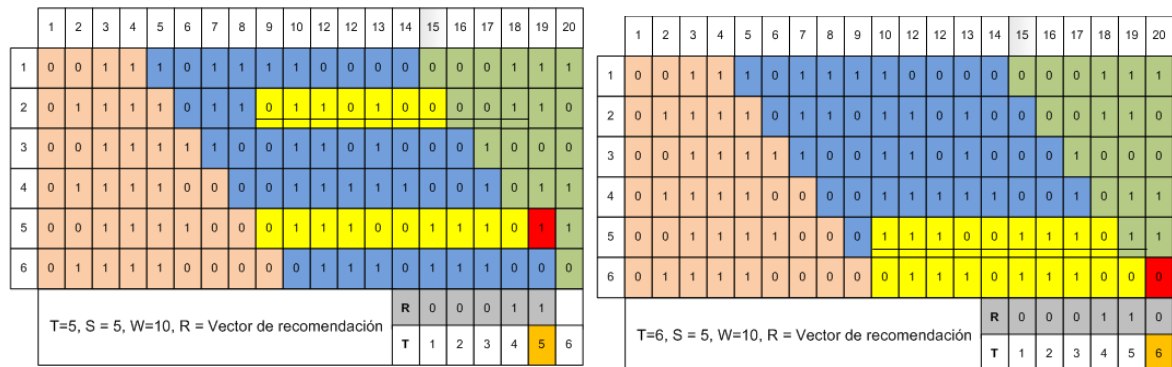
Figura 8.2: Ilustración del comportamiento del sistema dinámico de modelos.



(a) T1. Regla actual: 1. Regla candidata: 1. Mejor regla: 1. Recomendación: 1. (b) T2. Regla actual: 1. Regla candidata: 2. Mejor regla: 2. Recomendación: 0.



(c) T3. Regla actual: 2. Regla candidata: 3. Mejor regla: 2. Recomendación: 0. (d) T4. Regla actual: 2. Regla candidata: 4. Mejor regla: 2. Recomendación: 1.



(e) T5. Regla actual: 2. Regla candidata: 5. Mejor regla: 5. Recomendación: 1. (f) T6. Regla actual: 5. Regla candidata: 6. Mejor regla: 6. Recomendación: 1.

(g) En los períodos de 1 a 6. Conjunto de 5 modelos con un umbral de 4/5.

(h) Períodos de entrenamiento en color azul. Conjunto de modelos en amarillo. Períodos de prueba en color verde. Recomendaciones de inversión: "1" permanecer en el mercado, "0" permanecer en efectivo.

(i) Recomendaciones parciales de los elementos del sistema dinámico en rojo. Recomendación del sistema dinámico a continuación en el vector resultados R.

se mueve un período a la derecha y se genera la segunda regla, panel 8.2b, sobre la base del período de tiempos  $t_6 - t_{15}$ , (fila 2 amarillo). El rendimiento de esta regla candidata se compara con el rendimiento de la mejor regla en curso, regla 1, en el mismo período (marcado con una línea negra). Eso significa que consideraría 9 períodos de tiempo que formaron parte de su muestra de entrenamiento,  $t_6 - t_{14}$  (en amarillo), y uno de la muestra de prueba,  $t_{15}$  (en verde). A partir del torneo establecido entre ambas reglas, se determina que la mejor regla es la regla candidata número 2, que asume el rol de la mejor regla activa, y su recomendación, "0" se usaría como salida del sistema para el siguiente paso  $T2$ , que se corresponde con el segundo paso de tiempo del período de prueba  $t_{16}$  (en rojo) y el decimosexto elemento de la secuencia de la serie.

En  $T3$ , en 8.2c, la mejor regla en curso sería la segunda, y el nuevo candidato se obtendría a partir de  $t_7$  a  $t_{16}$  (fila 3 en amarillo). En el ejemplo, la comparación en este período favorecería la regla 2. Como resultado, la regla candidata, 3, se descartaría y, una vez más, la salida del sistema reflejaría la recomendación de la segunda regla  $t_{17}$  (en rojo) "0", quedando fuera del mercado. La dinámica del proceso en los períodos de prueba 4-6, representados en 8.2d-8.2f son muy similares. A medida que la ventana deslizante se mueve, genera nuevas reglas candidatas que se comparan con la mejor alternativa en ese momento, actualizándolas en consecuencia.

## 8.2 Análisis experimental

En esta sección, primero se describirá la configuración experimental, que incluye los siguientes elementos: los conjuntos de datos de entrenamiento y prueba utilizados, el protocolo experimental empleado, la parametrización utilizada, seguido de la presentación y discusión de los resultados.

### 8.2.1 Conjuntos de datos y diseño experimental

La evaluación experimental, de la solución propuesta, se realizará valorando su rendimiento frente a una serie de datos históricos, comparando los resultados alcanzados con otras estrategias equiparables que se apoyan, igualmente, en técnicas de GE, en particular las estrategias *Estática*, *Naif* y *Mayoría* descritas anteriormente en la sección 8.

Respecto a las muestras experimentales elegidas, en este estudio se trabajará, como en ocasiones anteriores, con dos conjuntos de datos financieros, que van desde el año 2013 hasta el 2017, que incluyen, por tanto, 5 años de datos diarios. El primero corresponde al índice Standard & Poor's 500, obtenido de Datastream, del que se extraen los datos de las sesiones bursátiles del índice, y el segundo, al rendimiento diario sin riesgos, obtenido del Banco de la Reserva Federal de Atlanta y disponible en <http://fred.stlouisfed.org/series/TB3MS>. La comparación de los tres métodos se basará en los retornos netos cosechados durante los años 2013, 2014, 2015, 2016 y 2017.

Teniendo en cuenta que las GE constituyen un método estocástico, los experimentos se realizarán 30 veces, tal como se expuso en el capítulo 5. La significación estadística de los resultados se probará formalmente utilizando el protocolo definido en el capítulo 5 *Metodología de evaluación*. y que por conveniencia resumimos a continuación:

Inicialmente, la normalidad de los retornos se evaluará mediante la prueba de *Kolmogorov-Smirnov*, aplicando la corrección de *Lilliefors*. En caso de que se rechace la normalidad de las observaciones se aplica la prueba de rangos de signos no paramétrica de *Wilcoxon*. De lo contrario la prueba de *Levene* comprueba la homocedasticidad de las varianzas. Si hay homocedasticidad se aplica la prueba *t* de *student*, en caso de que no la haya se aplicará la prueba *t* de *Welch*.

### 8.2.2 Parametrización

La parametrización general del algoritmo de las GE utilizado en esta tesis, está descrita, en más detalle, en el capítulo 4, mientras que la parametrización de la experimentación de este planteamiento, en particular, coincide en muchos aspectos con la realizada en el capítulo 7 sección 7.3.2. Se incluye, en esta sección, un resumen de los parámetros más importantes, con las particularidades referentes a la experimentación de la selección dinámica de modelos en inversión algorítmica.

La parametrización del algoritmo se estableció, tras la realización de algunos experimentos preliminares, como es el caso del número de reglas en liza del torneo, a los que se añadieron, para su uso, otros valores de parámetros comúnmente encontrados en la literatura.

Como tamaño de la población se eligió un valor de 500 individuos; se implementaron ciclos evolutivos de 50 generaciones y se incluyó la técnica de elitismo de un único individuo, en cada generación se seleccionó la mejor estrategia obtenida y se copió sin mutar en la población del siguiente ciclo evolutivo. Los individuos fueron inicializados usando series geométricas. En la inicialización de la población se estableció un valor de complejidad inicial mínimo de cinco y una probabilidad de crecimiento de 0,85.

Para mejorar la variabilidad en la población, y evitar una convergencia prematura, tanto durante el proceso de inicialización del algoritmo como durante la operación genética de mutación, si el mismo individuo (misma carga genética) aparecía más de una vez en la población, se realizaron hasta 100 intentos de reemplazarlo por uno nuevo.

El tamaño de la ventana de entrenamiento se estableció en tres años (753 sesiones), mientras que el de número de sesiones del período de prueba se fijó en un año (251 sesiones).

Por último, el número de reglas que componen el sistema de torneo ha sido de dos.

### 8.2.3 Resultados experimentales

Los resultados del experimento se presentan en la tabla 8.1. Allí se proporcionan las principales estadísticas descriptivas, de los retornos netos obtenidos en las muestras de prueba, de 30 experimentos realizados durante cinco años. Además de la aproximación *Dinámica*, que discutimos anteriormente, presentamos el rendimiento de otros tres sistemas con los que establecer una comparación: el sistema *Estático* estándar, donde se obtiene una regla única a partir de los tres años de entrenamiento más cercanos al período de prueba, y es entonces utilizado para todo el año de prueba; el que usa un modelo diferente por cada instante de tiempo, *Naif*; y la técnica de *Mayoría*, que se vale de la votación por mayoría simple para determinar el sentido de la recomendación.

Como podemos ver, el mejor rendimiento promedio se obtiene mediante la estrategia *Dinámica*. Si consideramos el rendimiento anual promedio de los cinco años completos, esta solución obtuvo un 10,71 % frente al 2,54 % de la aproximación

**Tabla 8.1:** Retorno neto en la comparativa de la estrategia dinámica.

	Estrategia	Media	Mediana	Var.	Max.	Min.	
2013	Dinámica	0,2161	0,2318	0,0017	0,2543	0,1311	
	Estática	0,0381	**	0,0409	0,0001	0,0715	0,0051
	Naif	0,1318	**	0,1318	0,0002	0,1520	0,0921
	Mayoría	0,1367	**	0,1363	0,0002	0,1557	0,0964
2014	Dinámica	0,0922	0,0994	0,0002	0,1042	0,0522	
	Estática	0,0606	**	0,0827	0,0015	0,1042	0,0079
	Naif	0,0559	**	0,0567	0,0002	0,0817	0,0257
	Mayoría	0,0852	**	0,0876	0,0001	0,0969	0,4710
2015	Dinámica	-0,0152	-0,0123	0,0002	-0,0123	-0,0935	
	Estática	-0,0160	*	-0,0123	0,0001	-0,0123	-0,0601
	Naif	-0,1728	**	-0,1785	0,0026	-0,0743	-0,2595
	Mayoría	-0,0903	**	-0,0817	0,0017	-0,0303	-0,1738
2016	Dinámica	0,0758	0,0889	0,0012	0,0889	-0,0217	
	Estática	0,0241	**	0,0043	0,0016	0,0889	-0,0223
	Naif	-0,1411	**	-0,1440	0,0014	-0,0335	-0,2193
	Mayoría	-0,0448	**	-0,0383	0,0011	0,0075	-0,1231
2017	Dinámica	0,1668	0,1668	0,0000	0,1668	0,1668	
	Estática	0,0204	**	0,0092	0,0016	0,1668	0,0092
	Naif	-0,0718	**	-0,0755	0,0011	0,0295	-0,1316
	Mayoría	0,0176	**	0,0294	0,0021	0,1094	-0,0753
Promedios	Dinámica	0,1071	0,1149	0,00066	0,1204	0,0470	
	Estática	0,0254	0,0250	0,00099	0,0838	-0,0120	
	Naif	-0,0396	-0,0419	0,00104	0,0311	-0,0985	
	Mayoría	0,0209	0,0267	0,00105	0,2026	0,0390	

- (a) \*\* Significación estadística vs. Dinámica al 1%. \* Significación estadística vs. Dinámica al 5%.
- (b) Incluye los gastos de transacción. Principales estadísticas descriptivas en 30 ejecuciones. Resultados de la prueba.
- (c) En color azul claro se marcan las técnicas y períodos en los que se obtienen los mejores rendimientos promedio.

*Estática* o el 2,09 % de la que se basa en el uso de una *Mayoría*. El sistema que mostró un peor rendimiento, con una pérdida neta de promedio anual del 3,96 %, proviene del uso de una ventana deslizante para entrenar reglas que sólo se usan una vez *Naif*. La estrategia *Dinámica* también produjo una buena consistencia y certeza en los resultados, ya que el promedio de las desviaciones anuales es muy bajo, del 0,00066, frente al 0,00099 de la aproximación *Estática*.

Una vez que analizamos los resultados año tras año, podemos ver que el modelo *Dinámico* domina claramente al *Estático*, a *Naif* y a *Mayoría*, no sólo en los años en que el mercado ofreció el peor rendimiento, como es el caso del 2015 y 2016, sino que también lo hizo en los años 2013, 2014 y el 2017. Aunque el rendimiento relativo entre *Estático*, *Naif* y *Mayoría* cambia entre ellos a lo largo del tiempo, *Dinámico* domina clara y consistentemente a los tres en todos los períodos considerados.

Señalar que la significación estadística de las diferencias de retorno medianas vs. *Dinámico* ha sido probada formalmente usando el protocolo del capítulo 5. Como podemos ver, todas las diferencias son significativas al 1 %.

Como en el experimento anterior, se pensó que las diferencias en los rendimientos de los tres métodos podrían explicarse básicamente a través del número de transacciones requeridas por las estrategias. De hecho, los datos proporcionados en la tabla 8.2 parecen evidenciar las discrepancias al respecto entre las tres soluciones presentadas.

Las dos estrategias que resultan en una menor cantidad de transacciones son la *Dinámica* y la *Estática*. En los años 2013, 2015 y 2016, la estrategia *Dinámica* ocupa el lugar más bajo de las dos, ambas empatan en el año 2015 y, finalmente, en 2017, es la *Estática* la que genera un menor número de transacciones. Por otro lado, con respecto a la estrategia de adaptación *Naif*, se observa que produce, consistentemente, muchas más órdenes de compra y venta que la *Dinámica* y la *Estática*. Este comportamiento debilitó seriamente su rentabilidad cuando se comparó con las otras dos estrategias y esto es más evidente una vez que consideremos el rendimiento bruto, sin costes de transacción. En cuanto a la estrategia *Mayoría*, es interesante observar que sus rendimientos netos son superiores a la estrategia *Estática* en los años 2013 y 2014, precisamente en los dos períodos en los que el número de transacciones en los que incurre son inferiores a los de esta, lo que reafirma la importancia de mantener bajos los costes de las operaciones.



Tabla 8.2: Número de transacciones estrategia dinámica.

	Estrategia	Media	Mediana	Var.	Min.	Max.
2013	Dinámico	4,93	4	6,82	2	10
	Estático	7,20	4	48,17	4	32
	Naif	14,00	14	16,00	4	20
	Mayoría	5,80	6	3,96	10	4
2014	Dinámico	4,40	4	5,63	2	10
	Estático	7,07	4	25,31	2	14
	Naif	14,73	14	11,72	8	22
	Mayoría	8,00	8	1,66	10	6
2015	Dinámico	2,40	2	3,42	2	12
	Estático	2,40	2	1,21	2	6
	Naif	42,40	42	24,94	36	56
	Mayoría	17,00	18	10,69	26	10
2016	Dinámico	2,73	2	8,69	2	18
	Estático	5,53	4	13,43	2	18
	Naif	72,47	72	58,40	62	92
	Mayoría	29,33	28	19,95	40	18
2017	Dinámico	2,00	2	0,00	2	2
	Estático	1,40	0	13,28	0	12
	Naif	79,40	81	102,39	52	100
	Mayoría	31,33	32	35,40	44	20
Promedio	Dinámico	3.29	2.80	4.91	2.00	10.40
	Estático	4.72	2.80	20.28	2.00	16.40
	Naif	44.60	44.60	42.69	42.00	48.40
	Mayoría	18.29	18.40	14.33	26.00	11.60

La tabla 8.3 representa los rendimientos brutos, es decir, los valores mostrados en la tabla 8.1 eliminando los costes de transacción. Si, nuevamente, calculamos los rendimientos promedio en los cinco años anteriores, vemos que la estrategia *Dinámica*, con un 11,56 %, aún ofrece los mejores resultados. A esto le sigue *Naif* con un 7,20 %, acompañada de cerca por la estrategia *Mayoritaria* con un 6,66 %, mientras que la aproximación tradicional *Estática* proporciona el peor rendimiento en términos brutos, 3,72 %.

Como podemos ver, la aproximación *Naif* ahora supera el rendimiento de la estrategia *Estática*, no sólo en el año 2013 sino también en el 2014 y el 2017. En el año 2016 el rendimiento obtenido por *Naif* se acerca hasta ser prácticamente igual al logrado por el sistema *Estático*. Es interesante observar que en el año 2017 *Naif* pasa de conseguir un retorno negativo a uno positivo, mientras que la solución *Estática* generó como resultado un número mayoritario de estrategias que no proporcionaron ninguna señal de compra y, por lo tanto, permanecieron fuera del mercado durante todo el período de prueba, ganando sólo el retorno de inversión sin riesgos.

En cuanto a *Mayoría*, esta queda también por debajo de *Naif* en los años 2013, 2016 y 2017. Por otra parte, hay una reversión de resultados cuando comparamos la estrategia *Estática*, que ofreció el segundo mejor rendimiento promedio en términos netos, con la *Mayoría* y la *Naif* que son las que obtienen el tercer y cuarto mejor lugar, respectivamente, en condiciones de rendimiento bruto. Probablemente, esto se debe al hecho de que *Naif* sigue de cerca la evolución del mercado y encuentra pequeñas oportunidades de negocio que podrían generar beneficios en caso de que no tuvieran que cubrir los costes de transacción.

Una vez más, el enfoque *Dinámico* parece ofrecer los resultados más fiables. No sólo el promedio de las varianzas de rendimiento anual, con un 0,00048, es menor en comparación con el 0,00088 de la segunda solución más estable, *Estática*, sino que también la estrategia es la mejor en términos de promedio de rendimiento mínimo, pues obtiene un 6,65 % frente al 2,11 % de la segunda mejor estrategia *Naif*.

Los resultados experimentales obtenidos apoyan la importancia de utilizar las soluciones con orientación *Dinámica*, porque tienen un gran potencial sobre las *Estáticas*, no sólo porque controlan los costes de transacción, que son claramente un factor importante, sino también porque ajustan mejor la estrategia de compra

**Tabla 8.3:** Retorno bruto en la comparativa de la estrategia dinámica.

	Estrategia	Media	Mediana	Var.	Max.	Min.
2013	Dinámico	0,2297	0,2531	0,0014	0,2593	0,1561
	Estático	0,0561	** 0,0509	0,0004	0,1215	0,0193
	Naif	0,1661	** 0,1654	0,0003	0,2215	0,1409
	Mayoría	0,1512	** 0,1519	0,0001	0,1676	0,1214
2014	Dinámico	0,1032	0,1092	0,0001	0,1141	0,0721
	Estático	0,0783	** 0,0927	0,0007	0,1092	0,0429
	Naif	0,0915	** 0,0890	0,0001	0,1165	0,0716
	Mayoría	0,1052	** 0,1074	0,0001	0,1184	0,0671
2015	Dinámico	-0,0092	-0,0073	0,0001	-0,0073	-0,0635
	Estático	-0,0100	-0,0073	0,0001	-0,0073	-0,0451
	Naif	-0,0627	** -0,0627	0,0024	0,0576	-0,1379
	Mayoría	-0,0478	** -0,0415	0,0014	0,0084	-0,1288
2016	Dinámico	0,0826	0,0939	0,0008	0,0939	-0,0042
	Estático	0,0379	** 0,0293	0,0015	0,0952	-0,0073
	Naif	0,0357	** 0,0409	0,0018	0,1346	-0,0282
	Mayoría	0,0285	** 0,0355	0,0010	0,0775	0,0481
2017	Dinámico	0,1718	0,1718	0,0000	0,1718	0,1718
	Estático	0,0239	** 0,0092	0,0017	0,1718	0,0092
	Naif	0,1296	** 0,1290	0,0013	0,1968	0,0591
	Mayoría	0,0959	** 0,0982	0,0014	0,1694	0,0323
Promedio	Dinámico	0,1156	0,1241	0,00048	0,1264	0,0665
	Estático	0,0372	0,0350	0,00088	0,0981	0,0038
	Naif	0,0720	0,0723	0,00119	0,1454	0,0211
	Mayoría	0,0666	0,0703	0,00081	0,1083	0,0280

- (a) \*\* Significación estadística vs. Dinámica al 1%. \* Significación estadística vs. Dinámica al 5%.
- (b) No incluye los costes de transacción. Principales estadísticas descriptivas en 30 ejecuciones. Resultados de la prueba.
- (c) En color azul claro se marcan las técnicas y períodos en los que se obtienen los mejores rendimientos promedio.

y venta. La estrategia *Dinámica* propuesta ofrece un buen compromiso entre estos dos objetivos en conflicto, que son ofrecer flexibilidad para adaptarse a los cambios estructurales sin sufrir por ello costes de penalización.

### 8.3 Resumen y conclusiones

El desarrollo de reglas de inversión, que sean flexibles, utilizando para ello las gramáticas evolutivas en su versión estándar, implica obtener una regla única basada en un período de entrenamiento, que luego se utiliza para generar recomendaciones a lo largo del tiempo. Por otro lado, el uso de técnicas de conjuntos de modelos mejora la adaptabilidad a los cambios estructurales que prevalecen en las series financieras, pero son muy sensibles a los costes de transacción. Por ello es necesario encontrar un sistema que ofrezca un equilibrio entre la flexibilidad para adaptarse a los cambios financieros, sin que al mismo tiempo incurra en excesivos costes de operación.

En este tercer experimento, se sugiere una nueva solución adaptativa que involucra un mecanismo de selección dinámico, que conmuta entre una regla activa y una regla candidata optimizada para los datos de mercado más recientes disponibles.

Para la evaluación experimental de la solución se han incluido otras tres alternativas similares, basadas en el mismo algoritmo central de las GE, durante un período de cinco años. Las estrategias comparadas incluyeron: la aproximación estática estándar *Estática*; una solución que usa un modelo diferente por cada instante de tiempo, *Naif*; la técnica de *Mayoría*, que se vale de la votación por mayoría simple para determinar el sentido de la recomendación; y los resultados de la solución propuesta *Dinámica* que se basan en mecanismo de ventana deslizante que establece una competencia entre reglas activas y candidatas para proporcionar las recomendaciones.

Los resultados obtenidos avalan la superioridad de la solución *Dinámica* en términos de retorno de inversión, seguida del enfoque *Estático*. El sistema que ofreció el peor rendimiento recayó en la estrategia *Naif*. La estrategia sugerida también mostró un mejor rendimiento en términos de previsibilidad.

El análisis del efecto de los costes de transacción en la rentabilidad alcanzada

pone de manifiesto la importancia de limitar el índice de compraventa, pues existe una clara relación inversa entre el número de órdenes de compra y venta y la rentabilidad conseguida. La estrategia *Naif* negocia mucho más a menudo que el resto de las otras soluciones, y las comisiones erosionan su rendimiento de manera muy significativa. Curiosamente, ese parece ser uno de los elementos de éxito en la solución *Dinámica* en comparación con la *Estática* y la *Naif* pero no el único. La solución dinámica sugiere la existencia de un mecanismo interno estable que proporciona un componente de histéresis, que evita una negociación excesiva, mientras, al mismo tiempo, se adapta a los cambios estructurales para optimizar sus decisiones de inversión.

Estos hallazgos avalan a la estrategia *Dinámica* como un mecanismo capaz de mantener el equilibrio buscado, entre la importancia de adaptarse a los cambios estructurales del mercado, y el objetivo de minimizar el riesgo que supone actualizar constantemente los modelos de recomendación.

*Esta página se ha dejado en blanco intencionadamente.*

## Capítulo 9

# Solución de selección dinámica vs. conjuntos de modelos

### 9.1 Introducción

En los dos experimentos anteriores se trataron dos soluciones distintas que perseguían el mismo objetivo, que se resume en buscar la deseada adaptación al estado del mercado, sin sufrir la penalización por los costes de transacción, derivados de una operativa que realiza una excesiva compra/venta. Cada uno de los experimentos produjo como resultado un sistema que prevalecía sobre el resto en el rendimiento obtenido.

Uno de los sistemas, que ha sido referido en las secciones anteriores como *Inercia*, utiliza un sistema de resistencia para controlar el cambio de la señal de compra/venta de la salida del conjunto. El mecanismo básico consiste en la selección de un conjunto de reglas cercanas al punto de predicción, que deciden la estrategia a seguir, entrar o salir del mercado, mediante un sistema de votación que emplea, en su decisión, una mayoría cualificada. La producción de las reglas se obtiene mediante las GE que utilizan, para su aprendizaje, una técnica de ventana deslizante sobre el conjunto de datos entrenamiento.

El segundo sistema, también se apoya en la técnica de ventanas deslizantes y el uso de las GE. No obstante, en esta ocasión, el enfoque adaptativo que se sugiere implica un mecanismo de selección dinámico, que conmuta entre una regla activa y una candidata que ha sido optimizada conforme a los datos de mercado más

recientes disponibles. La regla activa más cercana al punto de predicción decide la señal de compra/venta que se produce a la salida.

El estudio no estaría completo si no se realizase un análisis comparativo de los dos experimentos, examinando ambas soluciones en un ejercicio de comparación, para determinar cuál de los dos proporciona mejores rendimientos.

## 9.2 Análisis

Para proceder a la comparación se proporcionan el mismo tipo de tablas utilizadas en los dos experimentos anteriores.

**Tabla 9.1:** Retorno neto en la comparativa entre modelo basado en Inercia y el Dinámico.

	Estrategia	Media	Mediana	Var.	Max.	Min.
2013	Dinámico	0,2161	0,2318	0,0017	0,2543	0,1311
	Inercia	0,1423 **	0,1400	0,0006	0,1691	0,1298
2014	Dinámico	0,0922	0,0994	0,0002	0,1042	0,0522
	Inercia	0,0947	0,0938	0,0006	0,1048	0,0749
2015	Dinámico	-0,0152	-0,0123	0,0002	-0,0123	-0,0935
	Inercia	-0,0511 **	-0,0468	0,0006	-0,0214	-0,1327
2016	Dinámico	0,0758	0,0889	0,0012	0,0889	-0,0217
	Inercia	-0,0089 **	-0,0083	0,0011	0,0579	-0,0731
2017	Dinámico	0,1668	0,1668	0,0000	0,1668	0,1668
	Inercia	0,0615 **	0,0621	0,0009	0,1076	0,0065
Promedio	Dinámico	0,1071	0,1149	0,0007	0,1204	0,0470
	Inercia	0,0477	0,0482	0,0008	0,0836	0,0011

- (a) \*\* Significación estadística vs. Dinámico al 1%. \* Significación estadística vs. Dinámico al 5%.
- (b) Incluye los costes de transacción. Principales estadísticas descriptivas en 30 ejecuciones. Resultados de la prueba.
- (c) En color azul claro se marcan las técnicas y períodos en los que se obtienen los mejores rendimientos promedio.



En la tabla 9.1 vemos que la estrategia *Dinámica* es mejor que la basada en *Inercia* del conjunto de modelos, con significación estadística de un 1%, en los años 2013, 2015, 2016 y 2017. Merece la pena recalcar que los resultados de la estrategia *Dinámica* son decididamente mejores que los alcanzados con *Inercia*, por ejemplo en el 2016 *Inercia* tiene un comportamiento con pérdidas del 0,89% mientras que la *Dinámica* obtiene un rendimiento neto de aproximadamente un 7,5%. En el año 2014, por otra parte, al examinar los resultados de ambas estrategias no se puede descartar que sean estadísticamente iguales. Si, por otro lado, atendemos al promedio de los cinco años, la situación es más evidente. La solución *Dinámica*, con un 10,71%, más que dobla a su competidor, *Inercia*, que alcanza el 4,77%. Por otra parte las varianzas en ambas estrategias son muy bajas, aunque *Dinámica* en el 2017 presenta una varianza que indica que la regla elegida sigue consistentemente la misma estrategia.

**Tabla 9.2:** Número de transacciones comparativa entre modelo basado en *Inercia* y el *Dinámico*.

	Estrategia	Media	Mediana	Var.	Min.	Max.
2013	Dinámico	4,93	4	6,82	2	10
	Inercia	4,07	4	0,13	6	4
2014	Dinámico	4,40	4	5,63	2	10
	Inercia	4,93	5	2,96	8	2
2015	Dinámico	2,40	2	3,42	2	12
	Inercia	8,80	8	3,20	12	4
2016	Dinámico	2,73	2	8,69	2	18
	Inercia	11,00	12	3,79	14	8
2017	Dinámico	2,00	2	0,00	2	2
	Inercia	8,00	8	5,24	12	4
Promedio	Dinámico	3,29	2,80	4,91	2,0	10,40
	Inercia	7,36	7,40	3,06	10,40	4,40

Al igual que en pruebas anteriores, se ha procedido a comprobar cuál es el impacto, en términos de rendimientos, en las reglas de inversión derivado de los costes en las operaciones de compra y venta. En la tabla 9.2 es posible ver, que en los años 2013 y 2014 el número de operaciones de compra y venta de las dos

estrategias es básicamente igual, pero en el año 2013 no se traduce en una igualdad de rendimientos, sino que la estrategia *Dinámica* es más eficiente en el momento de decidir las entradas y salidas del mercado. En el año 2014 ambas estrategias entran y salen del mercado aproximadamente el mismo número de veces, siendo su rendimiento también parecido. Por otro lado, en los años 2015, 2016 y 2017 si hay una notable diferencia en el número de operaciones que ambas estrategias siguen, *Inercia* triplica e incluso cuadruplica a la estrategia *Dinámica*. El rendimiento en estos años favorece decididamente a la estrategia *Dinámica*, lo que parece indicar que hay una relación inversa entre órdenes de compra/venta y rentabilidad, sino fuera por los resultados del año 2013 que rompen ese nexo.

**Tabla 9.3:** Retorno bruto en la comparativa entre modelo basado en *Inercia* y el *Dinámico*.

	Estrategia	Media	Mediana	Var.	Max.	Min.
2013	Dinámico	0,2297	0,2531	0,0014	0,2593	0,1561
	Inercia	0,1525 **	0,1500	0,0001	0,1791	0,1398
2014	Dinámico	0,1032	0,1092	0,0001	0,1141	0,0721
	Inercia	0,1070	0,1067	0,0001	0,1248	0,0899
2015	Dinámico	-0,0092	-0,0073	0,0001	-0,0073	-0,0635
	Inercia	-0,0291 **	-0,0223	0,0001	-0,0014	-0,1077
2016	Dinámico	0,0826	0,0939	0,0008	0,0939	-0,0042
	Inercia	0,0186 **	0,0217	0,0010	0,0779	0,0381
2017	Dinámico	0,1718	0,1718	0,0000	0,1718	0,1718
	Inercia	0,0815 **	0,0818	0,0009	0,1260	0,0277
Promedio	Dinámico	0,1156	0,1241	0,00048	0,1264	0,0665
	Inercia	0,0661	0,0675	0,00044	0,1012	0,0375

(a) \*\* Significación estadística vs. *Dinámico* al 1%. \* Significación estadística vs. *Dinámico* al 5%.

(b) No incluye los costes de transacción. Principales estadísticas descriptivas en 30 ejecuciones. Resultados de la prueba.

(c) En color azul claro se marcan las técnicas y períodos en los que se obtienen los mejores rendimientos promedio.

La relación de rendimientos entre ambas estrategias, no cambia significativamente, una vez que son descontados los costes de operación, tabla 9.3. En el año

2013, 2015, 2016 y 2017, la estrategia *Dinámica* sigue siendo superior a la de *Inercia*. Por otro lado en el 2014, la mejora favorece a la estrategia *Dinámica* en relación a *Inercia*, que pasa a superarla con significación estadística. Si se examinan los promedios de los cinco años, la estrategia *Dinámica* con un 11,56 %, prácticamente dobla a la de *Inercia* con un 6,61 %. En cuanto a la robustez de los resultados, ambas técnicas se comportan prácticamente igual, con una varianza del 0,00048 de la solución *Dinámica* frente a un 0,00044 de la de *Inercia*.

Como conclusión se puede afirmar que, en este conjunto de cinco pruebas de períodos de un año, la técnica que pretende obtener un modelo de inversión, que cambia dinámicamente, es más poderosa que la que busca encontrar una solución a partir de conjuntos de modelos. La técnica *Dinámica* proporciona mejores rendimientos, cumpliendo al mismo tiempo mejor con los requisitos de adaptación al estado del mercado, sin penalización por una excesiva entrada/salida del mercado. Parece, por tanto, que su uso es muy prometedor.

*Esta página se ha dejado en blanco intencionadamente.*

## Capítulo 10

# Conclusiones y trabajos futuros

El uso de la CE, en el dominio de la generación y optimización de estrategias de inversión, tiene una larga historia. Una de las líneas de investigación, más populares, es la que se ha centrado en el uso de las técnicas de PG y GE, para desarrollar reglas de inversión basadas en indicadores técnicos. Entre las principales ventajas de este enfoque, podemos destacar la posibilidad de obtener reglas que sean flexibles, ya que la estructura no está completamente predefinida de antemano, y que sean simples e interpretables. Ambas técnicas, se enfrentan a un problema intrínseco a las series temporales financieras: las series contienen mucho ruido y están sujetas a cambios estructurales del mercado, que hacen extremadamente difícil extraer las características fundamentales que guíen a las estrategias de inversión.

Al concentrarnos, en primer lugar, en el estudio de la literatura, que versa sobre este tema, se ha visto que esta está basada en estudios con técnicas y configuraciones muy diferentes. Algunos autores, la mayoría, confían en el uso de la PG y otros en las GE; algunos consideran además extensiones como la adición de mecanismos de control de parsimonia o la implementación de técnicas de ADF, mientras que otros no. Dado que la mayoría de los estudios también se basan en muestras de series temporales financieras diferentes, la combinación de todos estos elementos hace que los esfuerzos de comparación sean muy desafiantes. Es difícil decir, a priori, si el algoritmo central de la (PG o de las GE) ofrece un mejor rendimiento, es más sencillo, flexible, robusto, que agregar las extensiones citadas tenga sentido, o incluso si su impacto es similar o asimétrico, dependiendo de la elección que hagamos. Por tanto, la primera parte de esta tesis ha estado dirigida, en primer lugar, a llenar este vacío en la literatura presentando un ejercicio exhaustivo de

evaluación comparativa. El resultado del estudio ha servido, también, para elegir aquella técnica que era la más apropiada, para cumplir con el objetivo de encontrar una adaptación al estado del mercado, sin sufrir la penalización de una excesiva compra/venta.

Para ello, este trabajo experimental ha comparado la rentabilidad y la complejidad alcanzada en las reglas de inversión generadas mediante el empleo de la PG, más específicamente la STGP y las GE, utilizando como entrada datos diarios de doce años del índice S&P 500, divididos en tres períodos consecutivos de cuatro años. Estas muestras se han diferenciado en el grado de dificultad para encontrar reglas de inversión, con el potencial de vencer una estrategia de referencia basada en B&H. Para cada uno de los algoritmos, se han probado cuatro configuraciones diferentes: el algoritmo central o básico; el algoritmo básico ampliado con control de parsimonia paramétrico lexicográfica; el algoritmo básico ampliado con técnicas de ADF y, finalmente, el algoritmo básico contando con ambas extensiones anteriores. Además de eso, se ha probado la sensibilidad de los resultados con poblaciones más grandes, y con un mayor número de indicadores técnicos formando parte del conjunto de funciones terminales. Para cumplir con este último requisito, se han agregado puntos de pivote y promedios móviles al conjunto de terminales sugerido por Lohpetch y Corne [121].

El rendimiento de los algoritmos ha dependido del período de tiempo seleccionado para el entrenamiento y prueba, pero han sido consistentes en el sentido de que las dificultades para encontrar reglas de inversión rentables, afectó a ambos algoritmos centrales de la misma manera. Si consideramos el resultado de los 192.000 experimentos realizados, ambos algoritmos ofrecieron retornos promedio que han sido muy similares. No obstante, las GE funcionaron relativamente mejor y también resultaron ser más fiables, en el sentido de que la proporción de estrategias que obtuvieron rendimientos iguales o mayores que las estrategia de referencia B&H fue mucho mayor. La estructura de los modelos de inversión obtenidos también fue diferente. Las GE construyeron individuos más simples, tanto en términos de profundidad promedio del árbol sintáctico que los representa, como del número de nodos del mismo, siendo, como consecuencia, mucho más fáciles de interpretar.

Con respecto a los beneficios de extender los algoritmos básicos, con el control de parsimonia paramétrico lexicográfica, las configuraciones que usaron este mecanismo de control de complejidad ofrecieron mejores rendimientos promedio, que

las que no hicieron uso del mismo. A pesar del hecho de que la afirmación anterior se aplica a ambos métodos de las GE y la PG, esta última tendía a construir árboles mayores y se benefició más de ello. El impacto de añadir a las ADF fue limitado en términos absolutos y, aunque mejoró el rendimiento promedio de las estrategias basadas en la PG, resultó tener un impacto negativo en las GE. Una razón puede ser debida, a que las soluciones ideales en las que participan las ADF, y en las que tienen mayor poder, son aquellas en las que los árboles de los individuos son de gran tamaño, con múltiples llamadas a subárboles dentro de ellos, aspecto que no sucede en las GE pero sí en la PG. Si fijamos la atención en la fiabilidad, ambos mecanismos proporcionaron valor añadido a las soluciones, ya que aumentaron la probabilidad de obtener resultados que fuesen al menos tan buenos como B&H. La contribución promedio de las extensiones, que se obtuvieron en los experimentos, fue la misma, pero la distribución de estas fue diferente. El control de parsimonia proporciona los mismos beneficios, tanto a la PG como a las GE, pero la primera absorbió la mayor parte de la contribución positiva del uso de las ADF.

Los resultados experimentales, de las pruebas de sensibilidad, mostraron que la extensión de la población inicial decrementó la solidez de las reglas basadas en las GE, y también resultó en una disminución media en la proporción de reglas basadas en la PG, que ofrecieron un retorno igual o superior a B&H. A pesar de esto, el retorno de promedio permaneció estable para la PG mientras que en las GE se vio ligeramente afectado de forma negativa. Esto aumenta la importancia de las distribuciones y la probabilidad de obtener estrategias especialmente malas.

Con respecto a los cambios en el conjunto de terminales, la adición de puntos de pivote y nuevos promedios móviles mejoró el rendimiento promedio de la PG, a costa de una disminución en las GE. Sin embargo, este cambio ayudó a ambos algoritmos básicos en términos de robustez de las estrategias de inversión, aunque no en la misma cantidad. La PG mostró mejores capacidades para explotar la nueva información.

Si se analizan los promedios totales de las técnicas de PG y GE, en todos los periodos, técnicas y análisis de sensibilidad efectuados, las GE presentan una mayor solidez, son más simples y proporcionan un retorno positivo por encima de B&H frente a uno negativo de la PG. Dicho esto las diferencias de rendimiento son muy pequeñas, no así las de solidez o niveles de complejidad alcanzados que favorecen ampliamente a las GE.

Por último, ante la pregunta sobre si los resultados obtenidos se podrían mejorar, utilizando diferentes conjuntos de parámetros y conjuntos de funciones, esta está aún abierta. Una razón para ello es que, dentro de la amplia gama de indicadores técnicos disponibles, los inversores tienden a adaptarse a las condiciones cambiantes del mercado, y eliminan, o descuentan, aquellos indicadores técnicos que cuando son utilizados generan pérdidas, o cuando, por otra parte, encuentran otros que resultaban ser más rentables. Por tanto, un estudio que determine que indicadores utilizar, en función del estado de mercado, es muy interesante, pues redundaría en la capacidad del AE empleado, para encontrar un modelo óptimo dentro del espacio de soluciones del problema. Tampoco se resuelve la pregunta de cuál de las dos técnicas de CE produce mejores rendimientos, puesto que a la vista de los resultados depende del periodo considerado y de los parámetros empleados en cada uno de ellos.

Por último, mencionar que el uso de las gramáticas resulto más flexible a la hora de definir los posibles arboles sintácticos de los individuos y el control de tipos, frente a la programación genética con control fuerte de tipos.

Es de esperar que estos resultados ayudarán a los futuros investigadores y profesionales, a tomar una decisión más informada con respecto a algunos de los componentes clave que se utilizarán en el desarrollo de sus sistemas de inversión, sobre la base del uso de técnicas de PG y GE. También se espera, que otros autores sigan el ejemplo, y repliquen el estudio en otros activos y períodos, para acumular evidencia más valiosa en diferentes condiciones de mercado. Eso, junto con un nuevo análisis sobre la sensibilidad a otros aspectos, como el uso de operadores específicos, establecimiento de límites en la complejidad o la parametrización de nuevas funciones técnicas, darían como resultado un progreso significativo hacia la finalización de esta imagen compleja.

Una vez determinado que la mejor opción de trabajo relativa residía en el uso de las GE, se decidió su uso para intentar resolver el problema del ajuste de los modelos de inversión a los cambios en las estructuras de mercado. En la primera parte de esta tesis se partió de la generación de reglas de inversión con representación flexible, utilizando las GE en su versión estándar. Esto implica obtener una única regla, en base a un período de entrenamiento, que posteriormente se utiliza para generar recomendaciones durante un determinado período de tiempo. Dada la prevalencia del cambio estructural en las series financieras, esto plantea un serio



obstáculo en la consecución de una solución robusta.

En un primer planteamiento, se utilizó un conjunto de reglas de inversión obtenidas mediante GE sobre una ventana deslizante. El sistema disponía de un componente crítico, el mecanismo de votación cualificado, que creó una inercia para evitar que el número de transacciones de compra/venta fuese excesivo, *Inercia*.

La evaluación experimental de esta solución involucró tres alternativas comparables apoyadas en el mismo algoritmo central durante un período de cinco años. Incluían una estrategia estática estándar, *Estática*; otra que empleaba un modelo diferente por cada paso de tiempo, *Naif*, y la utilización de un conjunto de estrategias para hacer recomendaciones, mediante el mecanismo de mayoría simple, *Mayoría*.

Los resultados apoyaron la superioridad del conjunto de modelos, con el componente de inercia, en términos de retorno de inversión, *Inercia*, seguido del enfoque *Estático* y del conjunto de modelos que implementa el voto mayoritario, *Mayoría*. El sistema de peor rendimiento fue la estrategia *Naif*. La estrategia sugerida también ofrecía ventajas en términos de disminución de la incertidumbre.

El análisis del impacto de los costes de transacción hizo evidente la importancia de limitar las transacciones comerciales, pues existía una clara relación inversa entre el número de órdenes de compra y venta y la rentabilidad. La estrategia *Naif* negoció mucho más a menudo que el resto de aproximaciones, y, por tanto, las comisiones erosionaron su rendimiento de manera muy significativa.

Por esta razón una política de inversión óptima ha de tener en cuenta este factor, y abstenerse de negociar si el estado de la cartera se encuentran dentro de una región sin transacción, y realizar transacciones al límite más cercano de la región, si el estado de la cartera se encuentran fuera de la misma. Debido a que los límites de las regiones sin transacción no se pueden derivar analíticamente, los algoritmos han de desarrollar reglas eficientes y simples que tengan en cuenta esos límites.

Estos hallazgos enfatizan la importancia de mantener un equilibrio entre la necesidad de adaptarse a los cambios estructurales y el riesgo de actualizar constantemente los modelos de recomendación de inversión, que están optimizados implícitamente para el largo plazo.

Si bien los resultados obtenidos por el sistema de modelos, que establecía un componente de inercia que se resistía al cambio de estado, eran prometedores, se

pensó que las propias reglas contaban con mecanismos implícitos de histéresis que al entrar en directa competición minimizarían el número de operaciones, mientras que aseguraban adaptarse a los cambios estructurales del mercado, de forma más eficiente que un sistema basado en votación con *Inercia*. Por tanto en un tercer planteamiento, se sugirió una nueva solución adaptativa (*Dinámica*) que involucraba un mecanismo de selección dinámico, que conmutase entre una regla activa y una regla candidata optimizada, para los datos de mercado más recientes disponibles.

Para la evaluación experimental de la solución, se incluyeron tres alternativas similares basadas en el mismo algoritmo central de las GE, durante un período de cinco años. Las estrategias comparadas incluyeron la aproximación estática estándar, *Estática*, la *Naif*, la *Mayoría* y los resultados de la solución propuesta *Dinámica*.

Los resultados obtenidos avalaron la superioridad de la solución *Dinámica* en términos de retorno de inversión, seguida del enfoque *Estático* y el de *Mayoría*. El sistema que ofreció el peor rendimiento recayó en la estrategia *Naif*. La estrategia sugerida también mostró un mejor rendimiento en términos de previsibilidad.

El análisis del efecto de los costes de transacción, en la rentabilidad alcanzada, puso de nuevo de manifiesto la importancia de limitar el índice de compraventa, con similares resultados que en el segundo planteamiento. Curiosamente, ese parece que fue uno de los elementos de éxito en la solución *Dinámica* en comparación con la *Estática*, la de *Mayoría*, y la *Naif* pero no el único. La solución dinámica, como era de esperar, sugirió la existencia de un mecanismo interno estable que proporciona un componente de histéresis, que evita una negociación excesiva, mientras, al mismo tiempo, se adapta a los cambios estructurales para optimizar sus decisiones de inversión.

Estos hallazgos avalan a la estrategia *Dinámica* como un mejor mecanismo capaz de cumplir con el objetivo de adaptarse a los cambios estructurales del mercado, al mismo tiempo que se minimizan los costes asociados a cambios constantes de modelos de recomendación.

El estudio se completó, finalmente, con una comparación entre las dos técnicas más prometedoras *Inercia* y *Dinámica*. Como conclusión a esa comparativa, es posible afirmar que la técnica *Dinámica* es más poderosa que la de *Inercia*, proporciona mejores rendimientos, y cumple al mismo tiempo mejor con los requisitos de adaptación al estado del mercado, sin penalización por una excesiva entrada/salida del mismo.

Una línea de trabajo futura, a partir de resultados de estos experimentos, consistiría en la expansión del análisis experimental a otros activos financieros o índices bursátiles diferentes, así como a instrumentos de inversión de renta fija. Dado que el problema central de la sobrecarga ha sido claramente identificado para las reglas basadas en las GE, factor que posiblemente se comparte con los que se obtendrían por la PG, es probable que la búsqueda de soluciones algorítmicas, que usen estas técnicas, sean adaptables y eviten transacciones excesivas, sea fructífera.

Una segunda línea de trabajo estaría dirigida a una investigación más profunda de los indicadores técnicos que son apropiados en el estado en el que el mercado se encuentra, que en si constituye otro estudio muy valioso, y los períodos temporales más ajustados a utilizar en el entrenamiento. En el análisis técnico no existen fórmulas mágicas ni estrategias infalibles, los indicadores que funcionaban en el pasado, dejan de hacerlo porque el mercado descuenta su uso. Aquí se ha trabajado con los más tradicionales, pero existen muchos otros como: la MACD (convergencia divergencia de la media móvil), el Oscilador Estocástico, Índice Direccional Medio o Índice de Movimiento Direccional, o la Divergencia que sería interesante investigar. Por último, dado que las series temporales financieras están sujetas a un ruido excesivo, sería interesante estudiar el incluir un proceso de filtrado de los datos de entrada, con el fin de facilitar al algoritmo la extracción de características.

*Esta página se ha dejado en blanco intencionadamente.*

## Capítulo 11

# Contribución de esta tesis

Se ha considerado importante dar cuenta del trabajo realizado, más allá de la investigación presentada en la tesis en sí. Como resultado se describe la producción científica del candidato durante su período doctoral marzo del 2017 a marzo de 2019, consistente en un artículo indexado en el Journal Citation Report.

**C. Martín, D. Quintana y P. Isasi.**

**Title:** Evolution of trading strategies with flexible structures: a configuration comparison [135].

**Date:** Dec 2019.

**Journal:** *Neurocomputing*.

**Volume:** 331.

**Pages:** 242-262.

**I.F. 2017:** 3.241 | Q1 Neurocomputing.

**doi:** 10.1016/J.NEUCOM.2018.11.062 <https://doi.org/10.1016/j.neucom.2018.11.062>.

**url:** <https://www.sciencedirect.com/science/article/pii/S0925231218314061?via%3Dihub>

**ISSN:** 0925-2312.

**Publisher:** Elsevier.

**Abstract:** Evolutionary Computation is often used in the domain of automated discovery of trading rules. Within this area, both Genetic Programming and Grammatical Evolution offer solutions with similar structures that have two key advantages in common: they are both interpretable and flexible in terms of their structure. The core algorithms can be extended to use automatically defined functions or mechanisms aimed to promote parsimony. The number of references on this topic is ample, but most of the studies focus on a specific setup. This means that it is not clear which is the best alternative. This work intends to fill that gap in the literature presenting a comprehensive set of experiments using both techniques with similar variations, and measuring their sensitivity to an increase in population size and composition of the terminal set. The experimental work, based on three S&P 500 data sets, suggest that Grammatical Evolution generates strategies that are more profitable, more robust and simpler, especially when a parsimony control technique was applied. As for the use of automatically defined function, it improved the performance in some experiments, but the results were inconclusive.

## Anexo A

# Selección de técnicas apropiadas de CE

En este apéndice, se recogen las mejores estrategias obtenidas en la configuración de línea base. En primer lugar, se muestran, por cada período de entrenamiento, las estrategias ordenadas por técnica, reflejando el valor de aptitud obtenido y la expresión “lisp” asociada al árbol del individuo. En la segunda parte, se presentan los gráficos en los que se puede ver los momentos temporales, en los que el algoritmo específico decide comprar y vender.

Las mejores estrategias por configuración y período, para experimento el base, se presentan en el apéndice A.1. Para aquellas configuraciones que usan la técnica de ADF se proporcionan tanto la regla principal, el árbol “0” del individuo, como el árbol ADF, ADF1. En algunos casos, aunque el algoritmo de PG evolucionó el árbol de ADF, este no fue empleado por el árbol principal del individuo y la regla no los utilizó, por lo tanto falta la referencia a ADF1 en la expresión. El comportamiento de estas reglas se ilustra en apéndice A.2. La línea más clara (verde), que aparece en las figuras, representa la recomendación de la regla. Si la recomendación es permanecer en el mercado esta línea sigue a la del índice, más oscura (roja), por el contrario, si aconseja al inversor salir de él la línea permanecerá plana en la parte inferior del gráfico.

### A.1 Mejores estrategias de inversión obtenidas en la configuración base

Mostramos los datos característicos del individuo citados anteriormente.

*Entrenamiento 2004-2006. Prueba 2007.*

- *Ge*  
 Fitness: 0.12847  
 Tree 0: (Or (>Mx2 M2) (Or (>Minimum UR) (<M10 LR)))
- *GePa*  
 Fitness: 0.12776  
 Tree 0: (Or (>Mx2 M2) (<M10 LR))
- *GeAdf*  
 Fitness: 0.12847  
 Tree 0: (Not (ADF1 (>M2 Mx2) (Or (>Minimum UR) (>LR M10))))  
 ADF1: (<ARG1 ARG0)
- *GeAdfPa*  
 Fitness: 0.11167  
 Tree 0: (ADF1 (>M2 Mx2)(Or (>Opening LR) (And (>M5 M2) (>M3 M2))))  
 Tree 1: (<ARG1 ARG0)
- *Gp*  
 Fitness: 0.14930  
 Tree 0: (Or (<Max M10) (Or (>M10 M5) (<M2 M5)))
- *GpPa*  
 Fitness: 0.14930  
 Tree 0: (Or (Or (<M3 M10) (<Max M10)) (Not (>M2 M5)))
- *GpAdf*  
 Fitness: 0.13329  
 Tree 0: (ADF1 (ADF1 (And (<Opening Closing) (<LR LR)) (<Minimum Minimum)) (And (Or (<M3 Minimum) (<UR Closing))(>Closing M3)))  
 ADF1: (And (Or (Not (<M5 M3)) (<M2 M5)) (<M2 Mx2))
- *GpAdfPa*  
 Fitness: 0.12776  
 Tree0: (ADF1 (<M2 Mx2) (>M10 M3))  
 ADF1: (Or ARG0 (<M10 LR))



*Entrenamiento 2008-2010. Prueba 2011.*

- *Ge*  
 Fitness: 0.04410  
 Tree 0: (And (<Min2 LR) (Not (And (>M10 LR) (Not (>Mx1 M3))))))
- *GePa*  
 Fitness: 0.07888  
 Tree 0: (And (Or (And (<M3 Minimum) (>M5 Closing)) (>LR Min2)) (>M2 Roc3))
- *GeAdf*  
 Fitness: 0.04410  
 Tree 0: (Or (<Mx1 M10) (Or (ADF1 (<M3 Mx1) (ADF1 (<LR Min1) (>Roc12 Min1))) (<M10 LR)))  
 Tree 1: (<ARG1 ARG0)
- *GeAdfPa*  
 Fitness: 0.10941  
 Tree 0: (Or (Not (<LR Min2)) (And (<M2 M3) (ADF1 (>M5 UR) (<LR Minimum))))))  
 ADF1: (<ARG0 ARG1)
- *Gp*  
 Fitness: 0.13300514044991257  
 Tree 0: (Or (And (>LR Min2) (Or (>LR Min2) (And (<UR M10) (Or (>Max Min2) (>M5 UR)))))(Or (Or (And (<UR M10) (Not (>M5 UR)))(>LR Min2)) (Or (And (Not (>M5 UR)) (>Roc12 Roc3))(And (>LR Min2)(>M5 UR))))))
- *GpPa*  
 Fitness: 0.13300  
 Tree 0: (Or (<Min1 LR) (And (Not (<UR M5))(Or (Or (<M3 Closing) (>Min2 M5)) (Or (<Roc3 Roc12) (>M10 UR))))))
- *GpAdf*  
 Fitness: 0.13300  
 Tree 0: (ADF1 (And (Or (<UR M10) (<Roc3 Roc12)) (<M5 UR)) (ADF1 (<Roc3

Roc12) (ADF1 (ADF1 (<UR M10)(ADF1 (<UR M10) (>Minimum Mx1))) (>Minimum Opening))))

ADF1: (Or (Not (<LR Min1)) (Or (<ARG0 ARG0) ARG0))

- *GpAdfPa*

Fitness: 0.17065

Tree 0: (Or (Or (ADF1 (<Max Min1) (<M10 Opening)) (<Min2 LR)) (Or (Not (<Min2 M5)) (<Min1 LR)))

ADF1: (And (And (<LR Minimum) (>Mx1 UR)) (And (Or (<Closing M5) (And (>ARG1 M10) (And ARG0 ARG0)))) (Or (<M5 UR) ARG0)))

*Entrenamiento 2012-2014. Prueba 2015.*

- *Ge*

Fitness: 0.02253

Tree 0: (And (Or (>Max M3) (And (Or (>Mx1 Roc3) (<M5 Min2)) (<Minimum Closing)))) (Or (>Max M3) (>Roc12 Roc3)))

- *GePa*

Fitness: 0.01009

Tree 0: (Or (>M5 Minimum) (<M5 M2))

- *GeAdf*

Fitness: 0.01009

Tree 0: (Or (<M5 M2) (>M5 Minimum))

ADF1: (>ARG1 ARG0)

- *GeAdfPa*

Fitness: 0.01009

Tree 0: (Or (<Minimum M5) (<M5 M2))

ADF1: (Or (<ARG1 ARG1) (<ARG1 ARG0))

- *Gp*

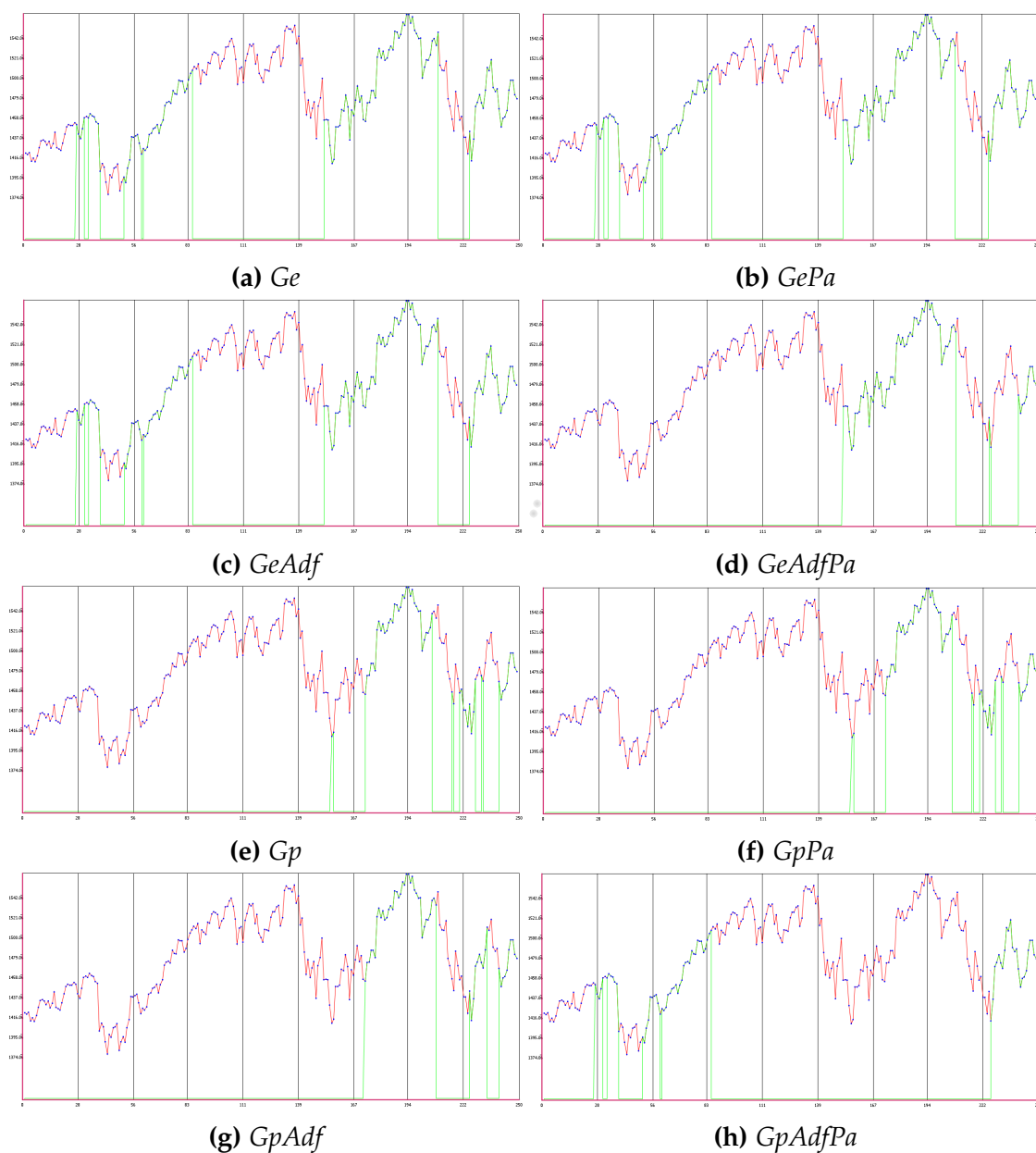
Fitness: 0.03046

Tree 0: (Or (And (<Minimum M5) (>M5 M3)) (Or (Or (And (<Minimum Mx2) (Or (<UR M5) (>Min2 UR)))) (Or (Not (>M5 M2)) (Or (<UR M5) (>Min2 UR)))) (<Max M5)))

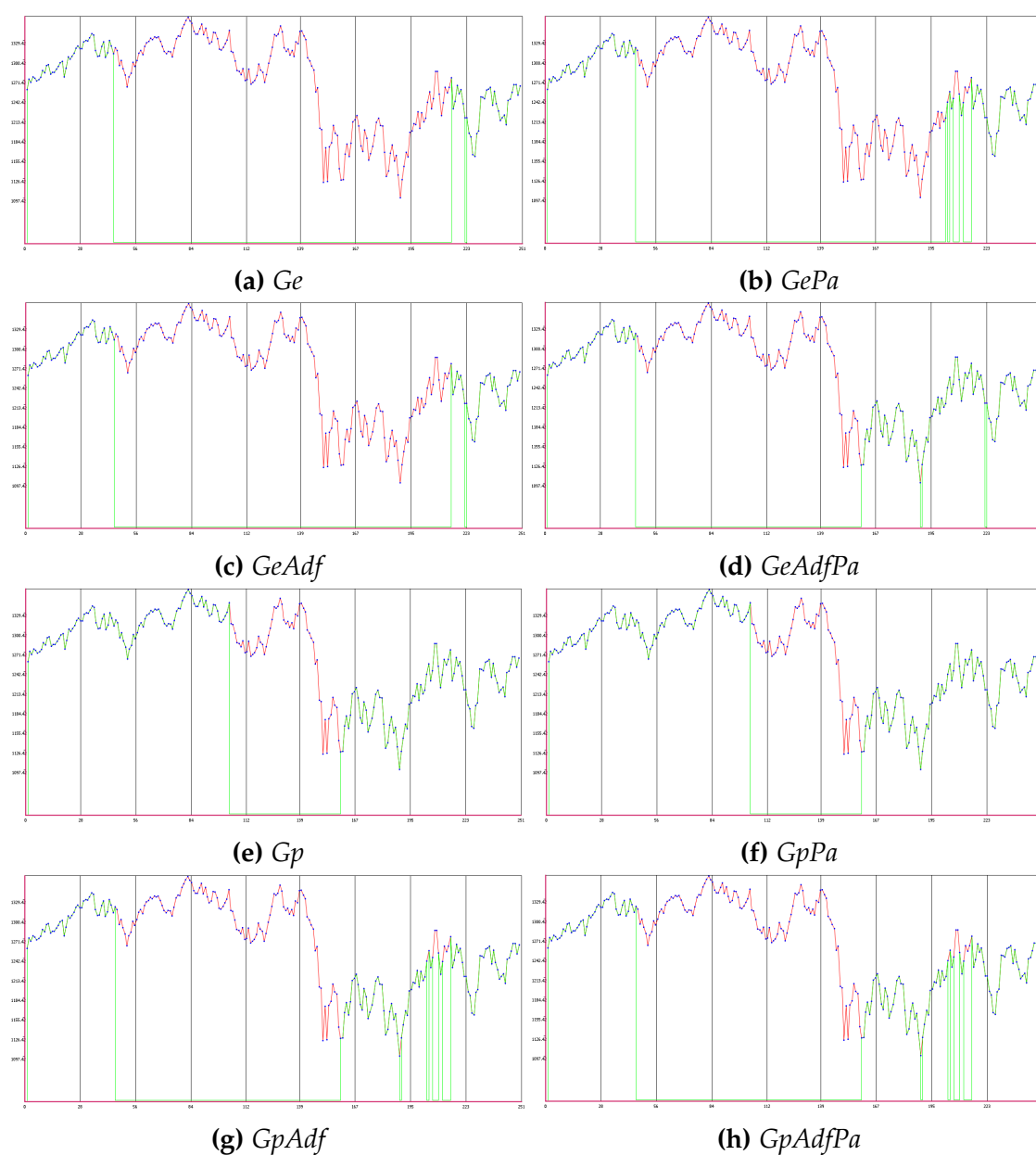
- *GpPa*  
 Fitness: 0.02252868979240462  
 Tree 0: (Or (Not (>M3 Opening)) (And (<Minimum Closing) (Not (>Roc3 Roc12))))
- *GpAdf*  
 Fitness: 0.026907  
 Tree 0: (Or (And (>M5 Opening) (Not (<M5 M3))) (Or (And (>M2 Opening) (Or (And (>M5 Opening) (>Mx2 Roc12)) (<M5 M2)))) (<M5 M2)))  
 ADF1: (Or (>ARG1 Minimum) (<ARG0 Max))
- *GpAdfPa*  
 Fitness: 0.02108  
 Tree 0: (Not (And (Not (Or (>M5 UR) (Not (>Max M5)))) (<M2 M5)))  
 ADF1: (And ARG1 (Or (<ARG0 Roc3) (And ARG1 ARG0)))

## A.2 Gráficos de las mejores estrategias de inversión obtenidas en la configuración base

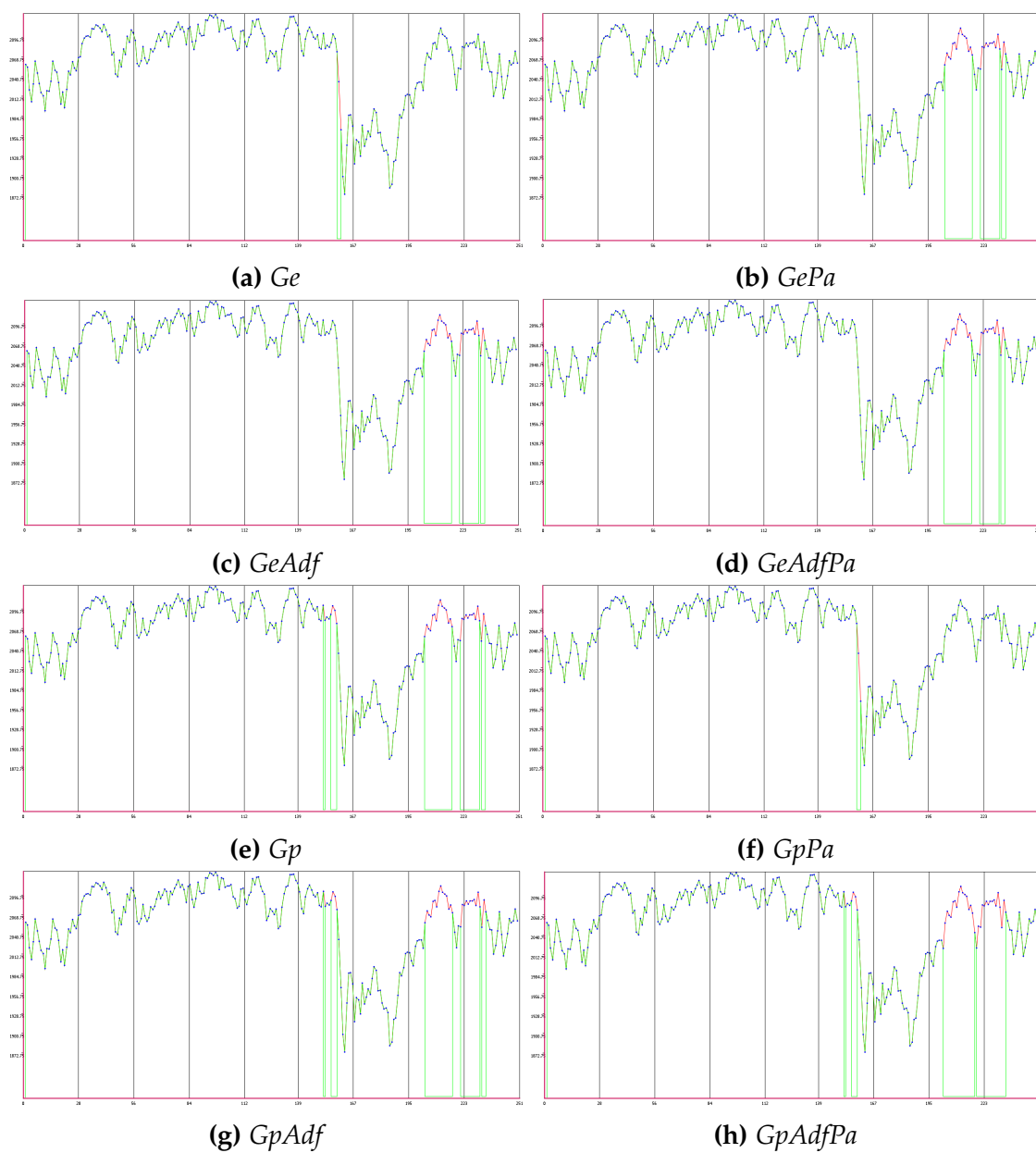
Los gráficos, que se muestran a continuación, se corresponden con los movimientos bursátiles en el año de prueba. El color más claro (verde) en el gráfico indica la recomendación de la regla, como una línea que marca los períodos en los que se está fuera o dentro del mercado.



**Figura A.1:** Comportamiento de las mejores estrategias configuración línea base. Entrenamiento: 2004-2006, prueba: 2007.



**Figura A.2:** Comportamiento de las mejores estrategias configuración línea base. Entrenamiento: 2008-2010, prueba: 2011.



**Figura A.3:** Comportamiento de las mejores estrategias configuración línea base. Entrenamiento: 2012-2014, prueba: 2015.

## Anexo B

# Protocolo de Pruebas

---

**Algoritmo 2:** Protocolo de significación estadística

---

Se aplica el test *Kolmogorov-Smirnov* con corrección *Lilliefors* para comprobar normalidad de las muestras

if P-valor resultante en alguna de las muestras es inferior al nivel de significación 0.05 se puede rechazar la hipótesis nula  $H_0$  de normalidad en las distribuciones **then**

    Aplicar el test de *Wilcoxon* para comparar las medianas de las muestras

    if P-valor resultante es inferior al nivel de significación 0,01 se puede rechazar la hipótesis nula  $H_0$  de que no hay diferencia entre los resultados de las muestras **then**

        if la mediana1 < mediana2 **then**

            Devolver “-”

**else**

            Devolver “++”

**end if**

**else if** P-valor resultante es inferior al nivel de significación 0,05 **then**

        if la mediana1 < mediana2 **then**

            Devolver “-”

**else**

            Devolver “+”

**end if**

**else**

        No se puede rechazar la hipótesis nula. Devolver “=”

**end if**

**else**

se acepta la normalidad de la distribución. aplicar el test de *Levene* para comprobar la igualdad de varianzas de las muestras

**if** P-valor resultante es inferior al nivel de significación 0,05 podemos rechazar la Hipótesis H0 igualdad de varianzas **then**

aplica el test de *Welch* a las muestras para comprobar la diferencia en las medias

**if** P-valor resultante es <al nivel de significación 0,01 podemos rechazar la hipótesis nula H0 de igualdad de medias **then**

**if** la media1 <media2 **then**

Devolver “-”

**else**

Devolver “++”

**end if**

**else if** el valor del P-valor resultante es inferior al nivel de significación 0,05 **then**

**if** la media1 <media2 **then**

Devolver “-”

**else**

Devolver “+”

**end if**

**else**

Se acepta la hipótesis nula de que las medias son iguales. Devolver “=”

**end if**

**else**

se acepta la hipótesis nula de que las varianzas son iguales Homocedasticidad

aplica el *t-test* a las muestras para comprobar la diferencia en las medias

**if** P-valor resultante es inferior al nivel de significación 0,01 podemos rechazar la hipótesis nula H0 de igualdad de medias **then**

**if** la media <media **then**

Devolver “-”

**else**

Devolver “++”

**end if**

**else if** el valor del P-valor resultante es inferior al nivel de significación 0,05 **then**

**if** la media1 <media2 **then**



```
        Devolver "-"
    else
        Devolver "+"
    end if
else
    Se acepta la hipótesis nula de que las medias son iguales. Devolver "="
end if
end if
end if
```

*Esta página se ha dejado en blanco intencionadamente.*

## Anexo C

# Selección técnica apropiada de CE

### C.1 Configuración base 500 individuos

#### C.1.1 Resultados de las simulaciones

ESTADISTICAS	ROI GE/GP TRADING RULES DATA SET 2004-2006, TEST 2007, 500 INDIVIDUOS, 50 GENERACIONES							
	GE	GE-PS	GE-ADF	GE-ADF-PS	GP	GP-PS	GP-ADF	GP-ADF-PS
$n$	2.000	2.000	2.000	2.000	2.000	2.000	2.000	2.000
$1-\alpha$	0,95	0,95	0,95	0,95	0,95	0,95	0,95	0,95
$z_{\alpha/2}$	1,96	1,96	1,96	1,96	1,96	1,96	1,96	1,96
$e$ MAXIMO	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05
$e = \frac{z_{\alpha/2} S}{\bar{x} \sqrt{n}}$	0,013736989	0,011235613	0,011730745	0,009816571	0,0187416	0,017246305	0,018144657	0,01769479
COMPLEJIDAD MEDIA	7,56	6,38	6,09	5,06	32,02	24,30	34,80	25,14
PROFUNDIDAD MEDIA	3,24	2,97	2,96	2,73	6,68	6,02	8,38	6,89
$\bar{x} = \frac{1}{n} \sum_{i=1}^{i=n} x_i$	0,04240245	0,04317808	0,04186491	0,04132803	0,03592355	0,03661806	0,03671493	0,03629149
% VECES MEJORA B&H Train.	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%
%MEJORA MEDIA Train.	10,11%	8,14%	5,92%	4,74%	42,95%	35,80%	35,17%	30,73%
%EMPEORA MEDIA Train.	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
% VECES MEJORA B&H Val.	63,70%	65,45%	55,05%	49,75%	53,60%	52,80%	50,15%	50,05%
%MEJORA MEDIA Val.	4,33%	6,60%	10,11%	11,88%	5,07%	7,93%	7,88%	4,27%
%EMPEORA MEDIA Val	-64,09%	-66,49%	-67,34%	-59,72%	-59,53%	-50,46%	-55,78%	-52,64%
% GUALA B&H	10,95%	14,10%	31,25%	40,35%	0,35%	1,00%	2,45%	2,55%
$s^2 = \frac{\sum_{i=1}^{i=n} (x_i - \bar{x})^2}{n-1}$	0,00017664	0,00012253	0,00012557	0,00008569	0,00000236	0,00020764	0,00023105	0,00021469
DESVIACION TIPICA $s$	0,01329050	0,01106927	0,01120559	0,00925685	0,01536190	0,01440955	0,01520024	0,01465241
$n = \left(\frac{z_{\alpha/2} S}{\bar{x} e}\right)^2$	151	101	110	77	281	238	263	250
$c_1 = \bar{x} - z_{\alpha/2} \frac{s}{\sqrt{n}}$	0,04181997	0,04269295	0,04137381	0,04092233	0,03525028	0,03598654	0,03604875	0,03564932
$c_2 = \bar{x} + z_{\alpha/2} \frac{s}{\sqrt{n}}$	0,04298493	0,04366322	0,04235602	0,04173373	0,03659681	0,03724959	0,03738111	0,03693366
COMPARACION	GPE/GPE-PS	GPE/GPE-ADF	GPE/GPE-ADF-PS	GPE/GP	GPE/GP-PS	GPE/GP-PS-ADF	GPE/GP-PS-ADF	
$\bar{x} = \bar{x}_1 - \bar{x}_2$	-0,00077563	0,00053754	0,00107442	0,00647890	0,00578439	0,00568752	0,00611096	
$s_x = \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}$	0,00000481	0,00000485	0,00000439	0,00000395	0,00000610	0,00000650	0,00000622	
$\bar{x} - z_{\alpha/2} s_x$	-0,00078505	0,00052804	0,00106582	0,00647116	0,00577244	0,00567478	0,00609878	
$\bar{x} + z_{\alpha/2} s_x$	-0,00076621	0,00054704	0,00108303	0,00648664	0,00579633	0,00570027	0,00612315	

Figura C.1: Resultados 2007. Configuración base (Entrenamiento 2004-06), 500 individuos.

ESTADÍSTICAS	ROI GE/GP TRADING RULES DATA SET 2008-2010, TEST 2011, 500 INDIVIDUOS, 50 GENERACIONES							
	GE	GE-PS	GE-ADF	GE-ADF-PS	GP	GP-PS	GP-ADF	GP-ADF-PS
$n$	2.000	2.000	2.000	2.000	2.000	2.000	2.000	2.000
$1-\alpha$	0,95	0,95	0,95	0,95	0,95	0,95	0,95	0,95
$z_{\alpha/2}$	1,96	1,96	1,96	1,96	1,96	1,96	1,96	1,96
$\epsilon$ MAXIMO	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05
$e = \frac{z_{\alpha/2} S}{\bar{x} \sqrt{n}}$	0,130484449	0,075654014	0,164640354	0,330413469	0,026727839	0,035706289	0,034550778	0,034619232
COMPLEJIDAD MEDIA	5,81	5,31	5,11	4,86	36,29	22,08	39,33	35,16
PROFUNDIDAD MEDIA	2,89	2,79	2,74	2,67	6,85	6,20	8,63	8,14
$\bar{x} = \frac{1}{n} \sum_{i=1}^{i=n} x_i$	0,00789402	0,01243441	-0,00812281	-0,00397126	-0,06333560	-0,04855223	-0,04806258	-0,04700329
% VECES MEJORA B&H Train.	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%
%MEJORA MEDIA Train.	403,96%	398,00%	385,40%	385,11%	686,99%	630,62%	616,97%	599,94%
%EMPEORA MEDIA Train.	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
% VECES MEJORA B&H Val.	84,80%	88,85%	74,70%	77,50%	28,90%	37,45%	37,20%	36,20%
%MEJORA MEDIA Val.	171,03%	172,66%	172,57%	172,21%	159,00%	161,96%	162,82%	161,12%
%EMPEORA MEDIA Val.	-483,75%	-497,09%	-563,18%	-559,54%	-529,22%	-517,61%	-500,14%	-495,73%
%IGUALA B&H	2,20%	1,65%	0,85%	1,05%	3,45%	7,35%	6,20%	8,50%
$s^2 = \sum_{i=1}^{i=n} \frac{(x_i - \bar{x})^2}{n-1}$	0,00055237	0,00046071	0,00093111	0,00089638	0,00001492	0,00156468	0,00143565	0,00137851
DESVIACIÓN TÍPICA $S$	0,02350260	0,02146427	0,03051417	0,02993954	0,03862519	0,03955605	0,03788993	0,03712827
$n = \left(\frac{z_{\alpha/2} S}{\bar{x} e}\right)^2$	13.621	4.579	21.685	87.338	572	1020	955	959
$c_1 = \bar{x} - z_{\alpha/2} \frac{S}{\sqrt{n}}$	0,00686397	0,01149370	-0,00946015	-0,00528342	-0,06502842	-0,05028585	-0,04972318	-0,04863051
$c_2 = \bar{x} + z_{\alpha/2} \frac{S}{\sqrt{n}}$	0,00892407	0,01337512	-0,00678547	-0,00265910	-0,06164277	-0,04681861	-0,04640198	-0,04537608
COMPARACION	GPE/GPE-PS	GPE/GPE-ADF	GPE-ADF-PS	GPE/GP	GPE/GP-PS	GPE/GP-PS-ADF	GPE/GP-PS-ADF	
$\bar{x} = \bar{x}_1 - \bar{x}_2$	-0,00454039	0,01601683	0,01186528	0,07122962	0,05644626	0,05595660	0,05489732	
$s_x = \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}$	0,00001608	0,00002421	0,00002354	0,00001236	0,00003710	0,00003440	0,00003321	
$\bar{x} - z_{\alpha/2} s_x$	-0,00457191	0,01596938	0,01181914	0,07120540	0,05637353	0,05588918	0,05483223	
$\bar{x} + z_{\alpha/2} s_x$	-0,00450886	0,01606428	0,01191143	0,07125384	0,05651898	0,05602402	0,05496240	

Figura C.2: Resultados 2011. Configuración base (Entrenamiento 2008-10), 500 individuos.

ESTADÍSTICAS	ROI GE/GP TRADING RULES DATA SET 2012-2014, TEST 2015, 500 INDIVIDUOS 50, GENERACIONES							
	GE	GE-PS	GE-ADF	GE-ADF-PS	GP	GP-PS	GP-ADF	GP-ADF-PS
$n$	2.000	2.000	2.000	2.000	2.000	2.000	2.000	2.000
$1-\alpha$	0,95	0,95	0,95	0,95	0,95	0,95	0,95	0,95
$z_{\alpha/2}$	1,96	1,96	1,96	1,96	1,96	1,96	1,96	1,96
$e$ MAXIMO	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05
$e = \frac{z_{\alpha/2}S}{\bar{x}\sqrt{n}}$	-0,032227957	-0,036742999	-0,039652317	-0,036984605	-0,025099995	-0,027610807	-0,027305641	-0,03204384
COMPLEJIDAD MEDIA	8,68	6,88	5,48	5,33	24,29	16,87	28,18	21,73
PROFUNDIDAD MEDIA	3,66	3,19	2,89	2,87	6,17	5,71	7,77	7,13
$\bar{x} = \frac{1}{n} \sum_{i=1}^{i=n} x_i$	0,01554141	0,01596802	0,01531471	0,01679303	0,01493168	0,01478448	0,01543992	0,01535530
% VECES MEJORA B&H Train.	100,00%	100,00%	100,00%	100,00%	100,00%	99,95%	100,00%	100,00%
%MEJORA MEDIA Train.	2,06%	1,06%	0,15%	0,06%	6,88%	3,25%	4,98%	2,75%
%EMPEORA MEDIA Train.	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
% VECES MEJORA B&H Val.	7,95%	4,55%	0,50%	0,30%	12,45%	6,95%	10,00%	6,65%
%MEJORA MEDIA Val.	184,80%	190,19%	196,09%	173,84%	138,08%	175,44%	155,09%	165,31%
%EMPEORA MEDIA Val	-77,53%	-73,36%	-55,43%	-70,55%	-186,44%	-65,29%	-78,28%	-70,03%
%I GUALA B&H	0,00%	0,00%	0,00%	0,00%	50,25%	0,00%	0,00%	0,00%
$s^2 = \sum_{i=1}^{i=n} \frac{(x_i - \bar{x})^2}{n-1}$	0,00013061	0,00017921	0,00019199	0,00020083	0,00000073	0,00008675	0,00009254	0,00012604
DESVIACIÓN TÍPICA $S$	0,01142831	0,01338704	0,01385595	0,01417128	0,00855148	0,00931417	0,00961959	0,01122695
$n = \left(\frac{z_{\alpha/2}S}{\bar{x}e}\right)^2$	831	1080	1258	1094	504	610	596	821
$c_1 = \bar{x} - z_{\alpha/2} \frac{s}{\sqrt{n}}$	-0,01604227	-0,01655474	-0,01592197	-0,01741412	-0,01530647	-0,01519270	-0,01586152	-0,01584734
$c_2 = \bar{x} + z_{\alpha/2} \frac{s}{\sqrt{n}}$	-0,01504054	-0,01538131	-0,01470744	-0,01617195	-0,01455690	-0,01437627	-0,01501833	-0,01486326
COMPARACION	GPE/GPE-PS	GPE/GPE-ADF	GPE-ADF-PS	GPE/GP	GPE/GP-PS	GPE/GP-PS-ADF	GPE/GP-PS-ADF	
$\bar{x} = \bar{x}_1 - \bar{x}_2$	-0,00042662	0,00022670	-0,00125163	0,00060972	0,00075692	0,00010148	0,00018611	
$s_x = \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}$	0,00000496	0,00000519	0,00000536	0,00000292	0,00000351	0,00000358	0,00000406	
$\bar{x} - z_{\alpha/2}S_x$	-0,00043633	0,00021652	-0,00126213	0,00060400	0,00075005	0,00009447	0,00017815	
$\bar{x} + z_{\alpha/2}S_x$	-0,00041690	0,00023688	-0,00124113	0,00061545	0,00076379	0,00010850	0,00019406	

Figura C.3: Resultados 2015. Configuración base (Entrenamiento 2012-14), 500 individuos.

## C.1.2 Estadísticos

Estadísticas dataset 2004-2006 test 2007 500-50 test					
	Media	Mediana	Var.	Max	Min
ge	0,042402451	0,054259118	0,000566278	0,12846835	-0,046632428
geAdf	0,041864912	0,054259118	0,000385003	0,12846835	-0,042719238
geAdfPa	0,041328028	0,031024829	0,000281876	0,1116699	-0,041298635
gePa	0,043178084	0,054259118	0,000485029	0,1277552	-0,052097704
gp	0,035923548	0,038083034	0,000770786	0,14930129	-0,067914054
gpAdf	0,036714927	0,031287234	0,000822004	0,13329124	-0,06555778
gpAdfPa	0,036291489	0,03109537	0,000724403	0,1277552	-0,0628708
gpPa	0,036618064	0,03863566	0,000652431	0,14930129	-0,049874637
Estadísticas dataset 2008-2010 test 2011 500-50 test					
	Media	Mediana	Var.	Max	Min
ge	0,007885443	0,025042504	0,001798853	0,044096522	-0,27549967
geAdf	-0,008139402	0,025042504	0,003475824	0,044096522	-0,22786741
geAdfPa	-0,003985775	0,025042504	0,003166991	0,10940749	-0,22786741
gePa	0,012454015	0,025042504	0,001435196	0,07887762	-0,22786741
gp	-0,063323436	-0,06029859	0,005409261	0,13300514	-0,3676943
gpAdf	-0,048062579	-0,039127324	0,005056933	0,13300514	-0,35010514
gpAdfPa	-0,047002345	-0,037981767	0,004787465	0,17064713	-0,3406693
gpPa	-0,048524954	-0,04131001	0,005116361	0,13300514	-0,35010514
Estadísticas dataset 2012-2014 test 2015 500-50 test					
	Media	Mediana	Var.	Max	Min
ge	-0,015544118	-0,010121487	0,000473546	0,02252869	-0,13847923
geAdf	-0,015317305	-0,010121487	4,14E-04	0,010093825	-0,13847923
geAdfPa	-0,016796372	-0,010121487	5,75E-04	0,010093825	-0,13847923
gePa	-0,015976988	-0,010121487	5,48E-04	0,010093825	-0,14798005
gp	-0,014921519	-0,010121487	0,0002842	0,030460501	-0,1551269
gpAdf	-0,015439923	-0,010121487	0,000335171	0,026906893	-0,15225029
gpAdfPa	-0,015354526	-0,010121487	0,000384786	0,021084607	-0,16864806
gpPa	-0,01479188	-0,010121487	0,000350104	0,02252869	-0,17333764

Figura C.4: Estadísticas configuración base 500 individuos.

Notas:

2004-2006 Rbh 0,0310248290.

2008-2010 Rbh -0,0171277130.

2012-2014 Rbh -0,0101214870.

(ge) Gramática evolutiva.

(gp) Programación genética.

(Adf) Funciones automáticamente definidas.

(Pa) Control de complejidad mediante parsimonia.

C.1.3 Contrastes estadísticos

Contrastes dataset 2004-2006 test 2007 500-50 test								
	ge	geAdf	geAdfPa	gePa	gp	gpAdf	gpAdfPa	gpPa
ge	NA	NA	NA	NA	NA	NA	NA	NA
geAdf	=	NA	NA	NA	NA	NA	NA	NA
geAdfPa	--	=	NA	NA	NA	NA	NA	NA
gePa	+	++	++	NA	NA	NA	NA	NA
gp	--	--	=	--	NA	NA	NA	NA
gpAdf	--	--	=	--	=	NA	NA	NA
gpAdfPa	--	--	=	--	=	=	NA	NA
gpPa	--	--	=	--	+	=	=	NA
Contrastes dataset 2008-2010 test 2011 500-50 test								
	ge	geAdf	geAdfPa	gePa	gp	gpAdf	gpAdfPa	gpPa
ge	NA	NA	NA	NA	NA	NA	NA	NA
geAdf	=	NA	NA	NA	NA	NA	NA	NA
geAdfPa	=	++	NA	NA	NA	NA	NA	NA
gePa	++	++	++	NA	NA	NA	NA	NA
gp	--	--	--	--	NA	NA	NA	NA
gpAdf	--	--	--	--	++	NA	NA	NA
gpAdfPa	--	--	--	--	++	=	NA	NA
gpPa	--	--	--	--	++	=	=	NA
Contrastes dataset 2012-2014 test 2015 500-50 test								
	geAdf	geAdfPa	gePa	gp	gpAdf	gpAdfPa	gpPa	
ge	NA	NA	NA	NA	NA	NA	NA	NA
geAdf	+	NA	NA	NA	NA	NA	NA	NA
geAdfPa	=	=	NA	NA	NA	NA	NA	NA
gePa	=	=	=	NA	NA	NA	NA	NA
gp	=	=	=	=	NA	NA	NA	NA
gpAdf	=	=	=	=	+	NA	NA	NA
gpAdfPa	=	=	=	=	++	++	NA	NA
gpPa	=	=	=	=	++	++	=	NA

Figura C.5: Contrastes estadísticos configuración base 500 individuos.

Notas:

(ge) Gramática evolutiva.

(gp) Programación genética.

(Adf) Funciones automáticamente definidas.

(Pa) Control de complejidad mediante parsimonia.

## C.2 Configuración población ampliada 3000 individuos

### C.2.1 Resultados de las simulaciones

ESTADÍSTICAS	ROI GE/GP TRADING RULES DATA SET 2004-2006, TEST 2007, 3000 INDIVIDUOS, 50 GENERACIONES							
	GE	GE-PS	GE-ADF	GE-ADF-PS	GP	GP-PS	GP-ADF	GP-ADF-PS
$n$	2000	2000	2000	2000	2000	2000	2000	2000
$1-\alpha$	0,95	0,95	0,95	0,95	0,95	0,95	0,95	0,95
$Z_{\alpha/2}$	1,96	1,96	1,96	1,96	1,96	1,96	1,96	1,96
$\epsilon$ MAXIMO	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05
$e = \frac{Z_{\alpha/2}S}{\bar{x}\sqrt{n}}$	0,01771237	0,01804231	0,01476761	0,01329671	0,02056058	0,01898684	0,01819554	0,01649132
COMPLEJIDAD MEDIA	12,50	10,64	9,22	7,47	37,21	34,13	42,05	33,89
PROFUNDIDAD MEDIA	4,47	4,07	3,81	3,32	6,86	6,71	9,09	8,23
$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$	0,03756654	0,03706292	0,04131210	0,04287858	0,03778716	0,03427715	0,04241606	0,04199896
% VECES MEJORA B&H Train.	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%
%MEJORA MEDIA Train.	23,68%	20,81%	15,16%	12,84%	61,78%	54,30%	51,06%	46,93%
%EMPEORA MEDIA Train.	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
% VECES MEJORA B&H Val.	43,05%	44,40%	60,95%	66,05%	55,25%	51,65%	60,10%	59,45%
%MEJORA MEDIA Val.	2,46%	1,90%	8,65%	9,02%	0,06%	9,10%	4,60%	0,43%
%EMPEORA MEDIA Val.	-44,64%	-48,41%	-50,94%	-52,91%	-68,88%	-61,60%	-63,01%	-56,63%
%I GUALA B&H	1,90%	1,80%	1,50%	1,55%	0,00%	0,00%	0,35%	0,10%
$s^2 = \sum_{i=1}^n \frac{(x_i - \bar{x})^2}{n-1}$	0,00023050	0,00023280	0,00019377	0,00016923	0,00000314	0,00022051	0,00031010	0,00024975
DESVIACIÓN TÍPICA $s$	0,01518227	0,01525776	0,01392023	0,01300899	0,01772714	0,01484965	0,01760979	0,01580348
$n = \left(\frac{Z_{\alpha/2}S}{\bar{x}e}\right)^2$	251	260	174	141	338	288	265	218
$c_1 = \bar{x} - Z_{\alpha/2} \frac{s}{\sqrt{n}}$	0,03690115	0,03639422	0,04070202	0,04230843	0,03701023	0,03362633	0,04164427	0,04130634
$c_2 = \bar{x} + Z_{\alpha/2} \frac{s}{\sqrt{n}}$	0,03823193	0,03773162	0,04192218	0,04344872	0,03856408	0,03492796	0,04318784	0,04269158
COMPARACION	GPE/GPE-PS	GPE/GPE-ADF	GPE-ADF-PS	GPE/GP	GPE/GP-PS	GPE/GP-PS-ADF	GPE/GP-PS-ADF	
$\bar{x} = \bar{x}_1 - \bar{x}_2$	0,00050362	-0,00374556	-0,00531204	-0,00022062	0,00328939	-0,00484952	-0,00443242	
$s_x = \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}$	0,00000733	0,00000673	0,00000639	0,00000515	0,00000713	0,00000864	0,00000760	
$\bar{x} - Z_{\alpha/2} s_x$	0,00048926	-0,00375876	-0,00532457	-0,00023072	0,00327541	-0,00486645	-0,00444732	
$\bar{x} + Z_{\alpha/2} s_x$	0,00051798	-0,00373237	-0,00529951	-0,00021052	0,00330337	-0,00483258	-0,00441753	

Figura C.6: Resultados 2007. Población ampliada 2007 (Entrenamiento 2004-06), 3.000 individuos.



ESTADÍSTICAS	ROI GE/GP TRADING RULES DATA SET 2008-2010, TEST 2011, 3000 INDIVIDUOS, 50 GENERACIONES							
	GE	GE-PS	GE-ADF	GE-ADF-PS	GP	GP-PS	GP-ADF	GP-ADF-PS
<b>n</b>	2.000	2.000	2.000	2.000	2.000	2.000	2.000	2.000
<b>1-<math>\alpha</math></b>	0,95	0,95	0,95	0,95	0,95	0,95	0,95	0,95
<b><math>z_{\alpha/2}</math></b>	1,96	1,96	1,96	1,96	1,96	1,96	1,96	1,96
<b>e MAXIMO</b>	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05
$e = \frac{z_{\alpha/2}S}{\bar{x}\sqrt{n}}$	-0,14886638	0,26189583	0,26067699	0,06286579	-0,02434406	-0,03120127	-0,02894699	-0,03018581
COMPLEJIDAD MEDIA	11,63	8,33	8,45	6,16	40,52	23,12	46,58	42,05
PROFUNDIDAD MEDIA	4,29	3,53	3,66	3,05	6,92	6,22	9,46	8,96
$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$	-0,00790000	0,00419186	0,00409356	0,01231526	-0,07296695	-0,04931657	-0,05780289	-0,05726817
% VECES MEJORA B&H Train.	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%
%MEJORA MEDIA Train.	466,33%	438,64%	432,27%	414,75%	881,32%	752,36%	759,99%	741,65%
%EMPEORA MEDIA Train.	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
% VECES MEJORA B&H Val.	68,40%	80,75%	80,50%	88,80%	23,55%	35,50%	33,75%	34,00%
%MEJORA MEDIA Val.	164,85%	168,56%	171,06%	171,21%	166,56%	164,10%	162,02%	160,60%
%EMPEORA MEDIA Val	-418,47%	-422,95%	-446,37%	-459,74%	-532,24%	-442,76%	-514,65%	-511,01%
%I GUALA B&H	5,60%	2,50%	3,15%	1,50%	0,00%	0,50%	0,80%	1,10%
$s^2 = \sum_{i=1}^n \frac{(x_i - \bar{x})^2}{n-1}$	0,00072005	0,00062746	0,00059282	0,00031206	0,00001643	0,00123267	0,00145755	0,00155579
DESVIACIÓN TÍPICA $S$	0,02683382	0,02504923	0,02434797	0,01766513	0,04053013	0,03510945	0,03817788	0,03944347
$n = \left(\frac{z_{\alpha/2}S}{\bar{x}e}\right)^2$	17.729	54.872	54.362	3.162	474	779	670	729
$c_1 = \bar{x} - z_{\alpha/2} \frac{s}{\sqrt{n}}$	-0,00907604	0,00309403	0,00302646	0,01154105	-0,07474326	-0,05085531	-0,05947611	-0,05899686
$c_2 = \bar{x} + z_{\alpha/2} \frac{s}{\sqrt{n}}$	-0,00672395	0,00528969	0,00516066	0,01308947	-0,07119063	-0,04777783	-0,05612967	-0,05553948
COMPARACION	GPE/GPE-PS	GPE/GPE-ADF	GPE-ADF-PS	GPE/GP	GPE/GP-PS	GPE/GP-PS-ADF	GPE/GP-PS-ADF	
$\bar{x} = \bar{x}_1 - \bar{x}_2$	-0,01209186	-0,01199356	-0,02021526	0,06506695	0,04141657	0,04990289	0,06146003	
$s_x = \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}$	0,00002136	0,00002086	0,00001755	0,00001611	0,00003192	0,00003635	0,00003751	
$\bar{x} - z_{\alpha/2} s_x$	-0,01213372	-0,01203444	-0,02024965	0,06503538	0,04135400	0,04983164	0,06138651	
$\bar{x} + z_{\alpha/2} s_x$	-0,01205000	-0,01195268	-0,02018087	0,06509851	0,04147914	0,04997414	0,06153355	

Figura C.7: Resultados 2011. Población ampliada (Entrenamiento 2008-10), 3.000 individuos.

ESTADÍSTICAS	ROI GE/GP TRADING RULES DATA SET 2012-2014, TEST 2015, 3000 INDIVIDUOS, 50 GENERACIONES							
	GE	GE-PS	GE-ADF	GE-ADF-PS	GP	GP-PS	GP-ADF	GP-ADF-PS
$n$	2.000	2.000	2.000	2.000	2.000	2.000	2.000	2.000
$1-\alpha$	0,95	0,95	0,95	0,95	0,95	0,95	0,95	0,95
$z_{\alpha/2}$	1,96	1,96	1,96	1,96	1,96	1,96	1,96	1,96
$\epsilon$ MAXIMO	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05
$e = \frac{z_{\alpha/2} S}{\bar{x} \sqrt{n}}$	0,03060759	0,03122381	0,03070347	0,03489979	0,02071508	0,02771977	0,02266313	0,02555675
COMPLEJIDAD MEDIA	8,61	6,67	7,00	5,86	32,57	21,69	37,90	27,57
PROFUNDIDAD MEDIA	3,66	3,16	3,24	3,00	6,75	5,95	8,75	7,65
$\bar{x} = \frac{1}{n} \sum_{i=1}^{i=n} x_i$	-0,01489995	-0,01569110	-0,01639352	-0,01668654	-0,01905746	-0,01653544	-0,01753718	-0,01659948
% VECES MEJORA B&H Train.	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%
%MEJORA MEDIA Train.	2,02%	0,96%	0,86%	0,40%	14,51%	9,39%	11,86%	8,98%
%EMPEORA MEDIA Train.	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
% VECES MEJORA B&H Val.	7,20%	3,55%	3,15%	1,25%	11,80%	14,60%	11,40%	12,80%
%MEJORA MEDIA Val.	140,01%	144,64%	141,81%	128,13%	139,60%	140,42%	135,43%	137,82%
%EMPEORA MEDIA Val	-307,69%	-411,90%	-405,23%	-488,22%	-186,07%	-212,06%	-179,71%	-208,68%
%IGUALA B&H	72,40%	80,75%	79,30%	84,30%	28,25%	42,30%	36,50%	44,95%
$s^2 = \sum_{i=1}^{i=n} \frac{(x_i - \bar{x})^2}{n-1}$	0,00010828	0,00012497	0,00013190	0,00017656	0,00000081	0,00010938	0,00008224	0,00009370
DESVIACIÓN TÍPICA $S$	0,01040574	0,01117888	0,01148467	0,01328765	0,00900763	0,01045838	0,00906857	0,00967963
$n = \left(\frac{z_{\alpha/2} S}{\bar{x} e}\right)^2$	749	780	754	974	343	615	411	523
$c_1 = \bar{x} - z_{\alpha/2} \frac{S}{\sqrt{n}}$	-0,01535601	-0,01618103	-0,01689685	-0,01726890	-0,01945223	-0,01699380	-0,01793463	-0,01702370
$c_2 = \bar{x} + z_{\alpha/2} \frac{S}{\sqrt{n}}$	-0,01444390	-0,01520116	-0,01589018	-0,01610418	-0,01866268	-0,01607708	-0,01713974	-0,01617525
COMPARACION	GPE/GPE-PS	GPE/GPE-ADF	GPE-ADF-PS	GPE/GP	GPE/GP-PS	GPE/GP-PS-ADF	GPE/GP-PS-ADF	
$\bar{x} = \bar{x}_1 - \bar{x}_2$	0,00079114	0,00149356	0,00178659	0,00415750	0,00163549	0,01753718	0,00169952	
$s_x = \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}$	0,00000370	0,00000382	0,00000463	0,00000242	0,00000344	0,00000184	0,00000320	
$\bar{x} - z_{\alpha/2} s_x$	0,00078390	0,00148608	0,00177751	0,00415276	0,00162874	0,01753358	0,00169325	
$\bar{x} + z_{\alpha/2} s_x$	0,00079839	0,00150104	0,00179566	0,00416225	0,00164223	0,01754079	0,00170580	

Figura C.8: Resultados 2015. Población ampliada (Entrenamiento 2012-14), 3.000 individuos.

## C.2.2 Estadísticos

Estadísticas dataset 2004-2006 test 2007 3000-50 test					
	Media	Mediana	Var.	Max	Min
ge	0,037566538	0,026108885	0,000762154	0,12877516	-0,051067024
geAd	0,041312103	0,054259118	0,000613261	0,14930129	-0,048365317
geAdfPa	0,042878577	0,054259118	0,000544185	0,12846835	-0,046632428
gePa	0,037062918	0,026108885	0,000765174	0,13495083	-0,051067024
gp	0,037787158	0,038709131	0,00101391	0,11960644	-0,06912166
gpAdf	0,042416054	0,040525995	0,000979322	0,16535366	-0,06912166
gpAdfPa	0,041998959	0,040525995	0,000882701	0,12574553	-0,07456678
gpPa	0,034277146	0,032804493	0,000781279	0,12627645	-0,0862634
Estadísticas dataset 2008-2010 test 2011 3000-50 test					
	Media	Mediana	Var.	Max	Min
ge	-0,007899999	0,025042504	0,002534867	0,07199911	-0,2528454
geAdf	0,00409356	0,025042504	0,002037397	0,13300514	-0,31833166
geAdfPa	0,012315261	0,025042504	0,00131646	0,044096522	-0,25027156
gePa	0,004191862	0,025042504	0,001956473	0,044096522	-0,282045
gp	-0,072966946	-0,07280269	0,005218934	0,16137011	-0,3559985
gpAdf	-0,057802888	-0,05519618	0,005336663	0,13713342	-0,38349068
gpAdfPa	-0,05726817	-0,05397049	0,005190618	0,13300514	-0,36063188
gpPa	-0,049316568	-0,05397049	0,004214433	0,13300514	-0,36133814
Estadísticas dataset 2012-2014 test 2015 3000-50 test					
Media	Media	Mediana	Var.	Max	Min
ge	-0,014902345	-0,010121487	0,000388032	0,02216942	-0,14018092
geAdf	-0,016396654	-0,010121487	5,15E-04	0,010093825	-0,170928
geAdfPa	-0,016674887	-0,010121487	5,52E-04	0,010093825	-0,13847923
gePa	-0,01567223	-0,010121487	4,87E-04	0,010093825	-0,17413168
gp	-0,019061926	-0,02078077	0,000278669	0,029541554	-0,13847923
gpAdf	-0,017537183	-0,011577831	0,000276319	0,02252869	-0,15114698
gpAdfPa	-0,016589667	-0,010121487	0,000324435	0,044906024	-0,16360565
gpPa	-0,01653865	-0,010121487	0,000307513	0,030460501	-0,17539094

Figura C.9: Estadísticas configuración extendida 3.000 individuos.

Notas:

2004-2006 Rbh 0,0310248290.

2008-2010 Rbh -0,0171277130.

2012-2014 Rbh -0,0101214870.

(ge) Gramática evolutiva.

(gp) Programación genética.

(Adf) Funciones automáticamente definidas.

(Pa) Control de complejidad mediante parsimonia.

## C.2.3 Contrastes estadísticos

Contrastes dataset 2004-2006 test 2007 3000-50 test								
	ge	geAdf	geAdfPa	gePa	gp	gpAdf	gpAdfPa	gpPa
ge	NA	NA	NA	NA	NA	NA	NA	NA
geAdf	=	NA	NA	NA	NA	NA	NA	NA
geAdfPa	--	=	NA	NA	NA	NA	NA	NA
gePa	+	++	++	NA	NA	NA	NA	NA
gp	--	--	=	--	NA	NA	NA	NA
gpAdf	--	--	=	--	=	NA	NA	NA
gpAdfPa	--	--	=	--	=	=	NA	NA
gpPa	--	--	=	--	+	=	=	NA
Contrastes dataset 2008-2010 test 2011 3000-50 test								
	ge	geAdf	geAdfPa	gePa	gp	gpAdf	gpAdfPa	gpPa
ge	NA	NA	NA	NA	NA	NA	NA	NA
geAdf	++	NA	NA	NA	NA	NA	NA	NA
geAdfPa	++	++	NA	NA	NA	NA	NA	NA
gePa	++	=	=	NA	NA	NA	NA	NA
gp	--	--	--	--	NA	NA	NA	NA
gpAdf	--	--	--	--	++	NA	NA	NA
gpAdfPa	--	--	--	--	++	=	NA	NA
gpPa	--	--	--	--	++	++	++	NA
Contrastes dataset 2012-2014 test 2015 3000-50 test								
	ge	geAdf	geAdfPa	gePa	gp	gpAdf	gpAdfPa	gpPa
ge	NA	NA	NA	NA	NA	NA	NA	NA
geAdf	=	NA	NA	NA	NA	NA	NA	NA
geAdfPa	=	=	NA	NA	NA	NA	NA	NA
gePa	=	+	=	NA	NA	NA	NA	NA
gp	--	--	--	--	NA	NA	NA	NA
gpAdf	--	--	--	--	++	NA	NA	NA
gpAdfPa	=	=	=	=	++	++	NA	NA
gpPa	=	=	=	=	++	++	=	NA

Figura C.10: Contrastes estadísticos configuración extendida 3.000 individuos.

Notas:

(ge) Gramática evolutiva.

(gp) Programación genética.

(Adf) Funciones automáticamente definidas.

(Pa) Control de complejidad mediante parsimonia.

### C.3 Configuración conjunto de funciones ampliado

#### C.3.1 Resultados experimento

ESTADÍSTICAS	ROI GE/GP TRADING RULES DATA SET 2004-2006, TEST 2007, 50 INDIVIDUOS, 50 GENERACIONES, FUNCTION SET EXTENDIDO							
	GE	GE-PS	GE-ADF	GE-ADF-PS	GP	GP-PS	GP-ADF	GP-ADF-PS
$n$	2.000	2.000	2.000	2.000	2.000	2.000	2.000	2.000
$1-\alpha$	0,95	0,95	0,95	0,95	0,95	0,95	0,95	0,95
$Z_{\alpha/2}$	1,96	1,96	1,96	1,96	1,96	1,96	1,96	1,96
$\epsilon$ MAXIMO	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05
$e = \frac{Z_{\alpha/2}S}{\bar{x}\sqrt{n}}$	0,01451680	0,01290200	0,01423623	0,01415687	0,01703076	0,01668308	0,01775122	0,01710391
COMPLEJIDAD MEDIA	6,79	5,69	5,75	4,76	34,48	30,36	34,57	29,20
PROFUNDIDAD MEDIA	3,15	2,87	2,88	2,65	6,78	6,54	8,04	7,21
$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$	0,04719753	0,04596351	0,04651781	0,04612925	0,05671059	0,05391140	0,05575719	0,05535540
% VECES MEJORA B&H Train.	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%
%MEJORA MEDIA Train.	23,88%	21,28%	15,46%	14,53%	86,67%	82,80%	74,53%	72,44%
%EMPEORA MEDIA Train.	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
% VECES MEJORA B&H Val.	72,45%	72,30%	59,10%	57,85%	71,65%	71,30%	71,20%	73,95%
%MEJORA MEDIA Val.	6,31%	13,56%	2,87%	2,35%	23,98%	18,53%	23,13%	15,95%
%EMPEORA MEDIA Val	-116,69%	-93,78%	-114,31%	-114,46%	-72,33%	-65,30%	-71,28%	-72,55%
%IGUALA B&H	21,50%	23,40%	38,00%	39,60%	0,40%	1,05%	1,65%	1,10%
$s^2 = \sum_{i=1}^n \frac{(x_i - \bar{x})^2}{n-1}$	0,00024440	0,00018309	0,00022832	0,00022203	0,00000486	0,00042114	0,00051001	0,00046669
DESVIACIÓN TÍPICA $s$	0,01563324	0,01353098	0,01511030	0,01490056	0,02203723	0,02052181	0,02258333	0,02160301
$n = \left(\frac{Z_{\alpha/2}S}{\bar{x}e}\right)^2$	169	133	162	160	232	223	252	234
$c_1 = \bar{x} - Z_{\alpha/2} \frac{s}{\sqrt{n}}$	0,04651237	0,04537049	0,04585557	0,04547621	0,05574477	0,05301199	0,05476744	0,05440861
$c_2 = \bar{x} + Z_{\alpha/2} \frac{s}{\sqrt{n}}$	0,04788268	0,04655653	0,04718005	0,04678230	0,05767642	0,05481080	0,05674695	0,05630220
COMPARACION	GPE/GPE-PS	GPE/GPE-ADF	GPE-ADF-PS	GPE/GP	GPE/GP-PS	GPE/GP-ADF	GPE/GP-PS-ADF	
$\bar{x} = \bar{x}_1 - \bar{x}_2$	0,00123401	0,00067972	0,00106827	-0,00951307	-0,00671387	-0,00855967	-0,00939189	
$s_x = \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}$	0,00000683	0,00000748	0,00000738	0,00000547	0,00001089	0,00001265	0,00001121	
$\bar{x} - Z_{\alpha/2}S_x$	0,00122063	0,00066506	0,00105380	-0,00952378	-0,00673521	-0,00858445	-0,00941386	
$\bar{x} + Z_{\alpha/2}S_x$	0,00124740	0,00069437	0,00108275	-0,00950236	-0,00669253	-0,00853488	-0,00936992	

Figura C.11: Resultados 2007. Funciones ampliadas (Entrenamiento 2004-06), 500 individuos.

ESTADÍSTICAS	ROI GE/GP TRADING RULES DATA SET 2008-2010, TEST 2011, 500 INDIVIDUOS, 50 GENERACIONES FS EXTENDED							
	GE	GE-PS	GE-ADF	GE-ADF-PS	GP	GP-PS	GP-ADF	GP-ADF-PS
<b>n</b>	2.000	2.000	2.000	2.000	2.000	2.000	2.000	2.000
<b>1-<math>\alpha</math></b>	0,95	0,95	0,95	0,95	0,95	0,95	0,95	0,95
<b><math>z_{\alpha/2}</math></b>	1,96	1,96	1,96	1,96	1,96	1,96	1,96	1,96
<b><math>e</math> MAXIMO</b>	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05
<b><math>e = \frac{z_{\alpha/2}S}{\bar{x}\sqrt{n}}</math></b>	-0,03836244	-0,03720716	-0,02686244	-0,02551614	-0,02949833	-0,02927582	-0,03196157	-0,03326623
COMPLEJIDAD MEDIA	5,91	5,65	5,47	5,04	35,93	33,49	36,69	32,42
PROFUNDIDAD MEDIA	2,94	2,88	2,85	2,73	6,83	6,74	8,35	7,77
<b><math>\bar{x} = \frac{1}{n} \sum_{i=1}^{i=n} x_i</math></b>	0,05556292	0,05745224	0,07846152	0,08342068	0,06796625	0,06631135	0,06192172	0,05679332
% VECES MEJORA B&H Train.	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%
%MEJORA MEDIA Train.	385,49%	381,44%	363,17%	361,58%	676,02%	666,21%	602,16%	598,99%
%EMPEORA MEDIA Train.	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
% VECES MEJORA B&H Val.	49,60%	48,00%	33,60%	31,95%	22,75%	22,55%	27,20%	28,20%
%MEJORA MEDIA Val.	172,43%	172,59%	173,06%	175,19%	179,98%	156,39%	171,13%	172,34%
%EMPEORA MEDIA Val.	49,40%	50,75%	65,20%	67,15%	64,95%	60,25%	56,75%	53,50%
%I GUALA B&H	1,00%	1,25%	1,20%	0,90%	12,30%	17,20%	16,05%	18,30%
<b><math>s^2 = \sum_{i=1}^{i=n} \frac{(x_i - \bar{x})^2}{n-1}</math></b>	0,00236538	0,00237895	0,00231272	0,00235882	0,00002093	0,00196206	0,00203920	0,00185832
Desviación típica <b>S</b>	0,04863515	0,04877445	0,04809070	0,04856771	0,04574564	0,04429513	0,04515752	0,04310819
<b><math>n = \left(\frac{z_{\alpha/2}S}{\bar{x}e}\right)^2</math></b>	1.177	1.107	577	521	696	686	817	885
<b><math>c_1 = \bar{x} - z_{\alpha/2} \frac{s}{\sqrt{n}}</math></b>	-0,05769445	-0,05958987	-0,08056919	-0,08554925	-0,06997114	-0,06825267	-0,06390084	-0,05868262
<b><math>c_2 = \bar{x} + z_{\alpha/2} \frac{s}{\sqrt{n}}</math></b>	-0,05343139	-0,05531460	-0,07635385	-0,08129210	-0,06596136	-0,06437003	-0,05994261	-0,05490402
COMPARACION	GPE/GPE-PS	GPE/GPE-ADF	GPE-ADF-PS	GPE/GP	GPE/GP-PS	GPE/GP-ADF	GPE/GP-PS-ADF	
$\bar{x} = \bar{x}_1 - \bar{x}_2$	-0,00188931	-0,02289860	-0,02785775	-0,01240333	-0,01074843	-0,00635880	0,00065891	
$s_x = \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}$	0,00007501	0,00007397	0,00007470	0,00005289	0,00006872	0,00006983	0,00006750	
$\bar{x} - z_{\alpha/2} s_x$	-0,00203634	-0,02304358	-0,02800416	-0,01250700	-0,01088312	-0,00649567	0,00052661	
$\bar{x} + z_{\alpha/2} s_x$	-0,00174228	-0,02275361	-0,02771135	-0,01229966	-0,01061374	-0,00622193	0,00079122	

Figura C.12: Resultados 2011. Funciones ampliadas (Entrenamiento 2008-10), 500 individuos.

ESTADÍSTICAS	ROI GE/GP TRADING RULES DATA SET 2012-2014, TEST 2015, 500 INDIVIDUOS, 50 GENERACIONES, FUNCTION SET EXTENDIDO							
	GE	GE-PS	GE-ADF	GE-ADF-PS	GP	GP-PS	GP-ADF	GP-ADF-PS
<b>n</b>	2.000	2.000	2.000	2.000	2.000	2.000	2.000	2.000
<b>1-<math>\alpha</math></b>	0,95	0,95	0,95	0,95	0,95	0,95	0,95	0,95
<b><math>Z_{\alpha/2}</math></b>	1,96	1,96	1,96	1,96	1,96	1,96	1,96	1,96
<b><math>e</math> MAXIMO</b>	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05
<b><math>e = \frac{Z_{\alpha/2}S}{\bar{x}\sqrt{n}}</math></b>	-0,03502082	-0,03402917	-0,03206175	-0,03819349	-0,02698227	-0,03198120	-0,02556883	-0,02691794
COMPLEJIDAD MEDIA	8,86	7,46	7,23	6,18	23,84	16,91	30,66	22,61
PROFUNDIDAD MEDIA	3,66	3,35	3,36	3,08	6,14	5,70	7,84	7,31
<b><math>\bar{x} = \frac{1}{n} \sum_{i=1}^{i=n} x_i</math></b>	-0,02056448	-0,01986335	-0,02039971	-0,01916587	-0,01503499	-0,01500671	-0,01559371	-0,01500775
% VECES MEJORA B&H Train.	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%
%MEJORA MEDIA Train.	6,23%	3,84%	2,60%	1,33%	6,68%	3,04%	24,16%	2,71%
%EMPEORA MEDIA Train.	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
% VECES MEJORA B&H Val.	1,40%	1,15%	1,40%	0,50%	11,20%	7,05%	2,10%	5,90%
%MEJORA MEDIA Val.	107,79%	98,00%	102,44%	116,98%	131,24%	132,38%	97,04%	132,12%
%EMPEORA MEDIA Val	-592,44%	-623,55%	-626,74%	-675,56%	-192,60%	-266,78%	-397,04%	-275,40%
%IGUALA B&H	80,00%	82,40%	81,30%	85,45%	53,10%	69,60%	83,25%	72,20%
<b><math>s^2 = \sum_{i=1}^{i=n} \frac{(x_i - \bar{x})^2}{n - 1}</math></b>	0,00027003	0,00023786	0,00022271	0,00027897	0,00000086	0,00011992	0,00008276	0,00008496
DESVIACIÓN TÍPICA <b>s</b>	0,01643248	0,01542278	0,01492348	0,01670232	0,00925636	0,01095063	0,00909744	0,00921756
<b><math>n = \left(\frac{Z_{\alpha/2}S}{\bar{x}e}\right)^2</math></b>	981	926	822	1167	582	818	523	580
<b><math>c_1 = \bar{x} - Z_{\alpha/2} \frac{s}{\sqrt{n}}</math></b>	-0,02128467	-0,02053928	-0,02105376	-0,01989788	-0,01544067	-0,01548665	-0,01599243	-0,01541172
<b><math>c_2 = \bar{x} + Z_{\alpha/2} \frac{s}{\sqrt{n}}</math></b>	-0,01984430	-0,01918741	-0,01974566	-0,01843386	-0,01462931	-0,01452678	-0,01519500	-0,01460377
COMPARACION	GPE/GPE-PS	GPE/GPE-ADF	GPE-ADF-PS	GPE/GP	GPE/GP-PS	GPE/GP-ADF	GPE/GP-PS-ADF	
<b><math>\bar{x} = \bar{x}_1 - \bar{x}_2</math></b>	-0,00070114	-0,00016477	-0,00139861	-0,00552950	-0,00555777	-0,00497077	-0,00485560	
<b><math>s_x = \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}</math></b>	0,00000805	0,00000783	0,00000868	0,00000604	0,00000661	0,00000632	0,00000565	
<b><math>\bar{x} - Z_{\alpha/2}S_x</math></b>	-0,00071691	-0,00018011	-0,00141563	-0,00554133	-0,00557072	-0,00498315	-0,00486667	
<b><math>\bar{x} + Z_{\alpha/2}S_x</math></b>	-0,00068537	-0,00014943	-0,00138160	-0,00551766	-0,00554482	-0,00495839	-0,00484453	

Figura C.13: Resultados 2015. Funciones ampliadas (Entrenamiento 2012-14), 500 individuos.

## C.3.2 Estadísticos

Estadísticas dataset 2004-2006 test 2007 500-50 FS extendido test					
	Media	Mediana	Var.	Max	Min
ge	0,047197525	0,042128168	0,000794127	0,1625893	-0,08153016
geAdf	0,046517809	0,042128168	0,000726536	0,1486748	-0,07151019
geAdfPa	0,046129251	0,042128168	0,000721903	0,1625893	-0,04295231
gePa	0,045963514	0,042128168	0,000598765	0,1514498	-0,05380335
gp	0,056710595	0,051690537	0,001686016	0,18853775	-0,07895879
gpAdf	0,055757193	0,051690537	0,001602927	0,18207785	-0,083910614
gpAdfPa	0,055355404	0,05125339	0,001492022	0,16885048	-0,09184138
gpPa	0,053911396	0,045242786	0,001474409	0,177595	-0,095166184
Estadísticas dataset 2008-2010 test 2011 500-50 extendido test					
	Media	Mediana	Var.	Max	Min
ge	-0,055562924	-0,017127713	0,008162094	0,13570783	-0,33114672
geAdf	-0,078461523	-0,110454	0,007372618	0,20198986	-0,33114672
geAdfPa	-0,083420677	-0,110454	0,00754161	0,13570783	-0,33114672
gePa	-0,057452238	-0,042414364	0,007982354	0,14379153	-0,33085275
gp	-0,06796625	-0,055548806	0,006546264	0,16023766	-0,4574118
gpAdf	-0,061921723	-0,037431818	0,006644842	0,1503476	-0,3680212
gpAdfPa	-0,056793323	-0,024516438	0,006424203	0,15321843	-0,3569802
gpPa	-0,066311353	-0,047714934	0,006423899	0,1481931	-0,35788494
Estadísticas dataset 2012-2014 test 2015 500-50 extendido test					
	Media	Mediana	Var.	Max	Min
ge	-0,020564485	-0,010121487	0,000877397	0,021281878	-0,18049368
geAdf	-0,020399713	-0,010121487	8,68E-04	0,011705417	-0,16171125
geAdfPa	-0,019165873	-0,010121487	8,10E-04	0,02252869	-0,16558762
gePa	-0,019863347	-0,010121487	8,41E-04	0,02045206	-0,17054616
gp	-0,015034988	-0,010121487	0,000284584	0,026906893	-0,16249278
gpAdf	-0,015593712	-0,010121487	0,000362195	0,03508383	-0,18946312
gpAdfPa	-0,015007746	-0,010121487	0,000343803	0,010093825	-0,17413168
gpPa	-0,015006713	-0,010121487	0,000348107	0,026906893	-0,14460951

Figura C.14: Estadísticas conjunto de funciones ampliado 500 individuos

Notas:

2004-2006 Rbh 0,0310248290.

2008-2010 Rbh -0,0171277130.

2012-2014 Rbh -0,0101214870.

(ge) Gramática evolutiva.

(gp) Programación genética.

(Adf) Funciones automáticamente definidas.

(Pa) Control de complejidad mediante parsimonia.



### C.3.3 Contrastes estadísticos

Contrastes dataset 2004-2006 test 2007 500-50 extendido test								
	ge	geAdf	geAdfPa	gePa	gp	gpAdf	gpAdfPa	gpPa
ge	NA	NA	NA	NA	NA	NA	NA	NA
geAdf	=	NA	NA	NA	NA	NA	NA	NA
geAdfPa	=	=	NA	NA	NA	NA	NA	NA
gePa	=	++	++	NA	NA	NA	NA	NA
gp	++	++	++	++	NA	NA	NA	NA
gpAdf	++	++	++	++	=	NA	NA	NA
gpAdfPa	++	++	++	++	=	=	NA	NA
gpPa	++	++	++	++	--	-	-	NA
Contrastes dataset 2008-2010 test 2011 500-50 extendido test								
	ge	geAdf	geAdfPa	gePa	gp	gpAdf	gpAdfPa	gpPa
ge	NA	NA	NA	NA	NA	NA	NA	NA
geAdf	--	NA	NA	NA	NA	NA	NA	NA
geAdfPa	--	=	NA	NA	NA	NA	NA	NA
gePa	=	++	++	NA	NA	NA	NA	NA
gp	--	++	++	--	NA	NA	NA	NA
gpAdf	--	++	++	=	++	NA	NA	NA
gpAdfPa	--	++	++	=	++	+	NA	NA
gpPa	--	++	++	--	=	--	--	NA
Contrastes dataset 2012-2014 test 2015 500-50 extendido test								
	ge	geAdf	geAdfPa	gePa	gp	gpAdf	gpAdfPa	gpPa
ge	NA	NA	NA	NA	NA	NA	NA	NA
geAdf	=	NA	NA	NA	NA	NA	NA	NA
geAdfPa	++	+	NA	NA	NA	NA	NA	NA
gePa	=	=	=	NA	NA	NA	NA	NA
gp	=	=	=	=	NA	NA	NA	NA
gpAdf	++	++	=	++	++	NA	NA	NA
gpAdfPa	=	=	=	=	++	=	NA	NA
gpPa	=	=	=	=	++	=	=	NA

Figura C.15: Contrastes conjunto de funciones ampliado 500 individuos.

Notas:

(ge) Gramática evolutiva.

(gp) Programación genética.

(Adf) Funciones automáticamente definidas.

(Pa) Control de complejidad mediante parsimonia.

## C.4 Configuración conjunto de funciones y población ampliada 3.000 individuos

### C.4.1 Resultados de las simulaciones

ESTADÍSTICAS	ROI GE/GP TRADING RULES DATA SET 2004-2006, TEST 2007, 3000 INDIVIDUOS, 50 GENERACIONES, FUNCTION SET EXTENDIDO							
	GE	GE-PS	GE-ADF	GE-ADF-PS	GP	GP-PS	GP-ADF	GP-ADF-PS
$n$	2.000	2.000	2.000	2.000	2.000	2.000	2.000	2.000
$1-\alpha$	0,95	0,95	0,95	0,95	0,95	0,95	0,95	0,95
$z_{\alpha/2}$	1,96	1,96	1,96	1,96	1,96	1,96	1,96	1,96
$e$ MAXIMO	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05
$e = \frac{z_{\alpha/2}S}{\bar{x}\sqrt{n}}$	0,01685503	0,01638928	0,01558317	0,01421555	0,01514938	0,01512634	0,01597366	0,01513813
COMPLEJIDAD MEDIA	9,73	8,49	7,57	6,20	38,24	37,51	40,67	37,24
PROFUNDIDAD MEDIA	3,89	3,64	3,36	3,10	6,85	6,82	8,77	8,36
$\bar{x} = \frac{1}{n} \sum_{i=1}^{i=n} x_i$	0,05495578	0,05375038	0,04967188	0,04879378	0,05992795	0,05703522	0,06023778	0,06018979
% VECES MEJORA B&H Train.	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%
%MEJORA MEDIA Train.	45,68%	41,56%	32,43%	29,90%	108,21%	108,67%	96,51%	95,35%
%EMPEORA MEDIA Train.	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
% VECES MEJORA B&H Val.	87,85%	89,95%	88,60%	89,70%	76,60%	72,55%	76,10%	74,75%
%MEJORA MEDIA Val.	6,35%	1,69%	7,78%	11,39%	25,14%	20,04%	26,87%	26,06%
%EMPEORA MEDIA Val.	-141,28%	-158,88%	-159,00%	-151,13%	-71,59%	-60,83%	-69,35%	-63,39%
%IGUALA B&H	0,05%	0,15%	4,30%	4,15%	0,00%	0,00%	0,00%	0,00%
$s^2 = \frac{\sum_{i=1}^{i=n} (x_i - \bar{x})^2}{n-1}$	0,00044669	0,00040402	0,00031193	0,00025048	0,00000429	0,00038750	0,00048202	0,00043222
DESVIACIÓN TÍPICA $s$	0,02113499	0,02010020	0,01766141	0,01582658	0,02071492	0,01968502	0,02195494	0,02078998
$n = \left(\frac{z_{\alpha/2}S}{\bar{x}e}\right)^2$	227	215	194	162	184	183	204	183
$c_1 = \bar{x} - z_{\alpha/2} \frac{s}{\sqrt{n}}$	0,05402950	0,05286945	0,04889784	0,04810015	0,05902008	0,05617249	0,05927556	0,05927863
$c_2 = \bar{x} + z_{\alpha/2} \frac{s}{\sqrt{n}}$	0,05588207	0,05463131	0,05044593	0,04948741	0,06083582	0,05789795	0,06119999	0,06110095
COMPARACION	GPE/GPE-PS	GPE/GPE-ADF	GPE-ADF-PS	GPE/GP	GPE/GP-PS	GPE/GP-ADF	GPE/GP-PS-ADF	
$\bar{x} = \bar{x}_1 - \bar{x}_2$	0,00120540	0,00528390	0,00616200	-0,00497217	-0,00207944	-0,00528199	-0,00643941	
$s_x = \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}$	0,00001347	0,00001218	0,00001145	0,00000999	0,00001322	0,00001469	0,00001323	
$\bar{x} - z_{\alpha/2}S_x$	0,00117901	0,00526002	0,00613956	-0,00499175	-0,00210535	-0,00531079	-0,00646534	
$\bar{x} + z_{\alpha/2}S_x$	0,00123180	0,00530778	0,00618444	-0,00495259	-0,00205352	-0,00525319	-0,00641348	

Figura C.16: Resultados 2007. Funciones ampliadas (Entrenamiento 2004-06), 3.000 individuos.

ESTADÍSTICAS	ROI GE/GP TRADING RULES DATA SET 2008-2010, TEST 2011, 3000 INDIVIDUOS, 50 GENERACIONES FS EXTENDED							
	GE	GE-PS	GE-ADF	GE-ADF-PS	GP	GP-PS	GP-ADF	GP-ADF-PS
<b>n</b>	2.000	2.000	2.000	2.000	2.000	2.000	2.000	2.000
<b>1-<math>\alpha</math></b>	0,95	0,95	0,95	0,95	0,95	0,95	0,95	0,95
<b><math>Z_{\alpha/2}</math></b>	1,96	1,96	1,96	1,96	1,96	1,96	1,96	1,96
<b><math>e</math> MAXIMO</b>	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05
<b><math>e = \frac{Z_{\alpha/2}S}{\bar{x}\sqrt{n}}</math></b>	-0,13620729	-0,43426932	-0,26192379	-0,51941747	-0,02746486	-0,03175578	-0,03405925	-0,03331261
COMPLEJIDAD MEDIA	8,39	6,88	6,59	5,38	39,62	23,00	41,28	38,53
PROFUNDIDAD MEDIA	3,51	3,15	3,14	2,85	6,89	6,21	8,89	8,54
<b><math>\bar{x} = \frac{1}{n} \sum_{i=1}^{i=n} x_i</math></b>	0,01171868	0,00343872	0,00611021	0,00293331	0,06867304	0,05537202	0,05684774	0,05784357
% VECES MEJORA B&H Train.	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%
%MEJORA MEDIA Train.	440,02%	424,22%	413,29%	405,22%	829,88%	753,03%	720,06%	720,94%
%EMPEORA MEDIA Train.	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
% VECES MEJORA B&H Val.	73,50%	78,80%	77,70%	80,65%	23,05%	27,50%	29,50%	26,45%
%MEJORA MEDIA Val.	174,41%	172,98%	172,77%	172,48%	183,92%	186,66%	166,32%	177,15%
%EMPEORA MEDIA Val	-663,19%	-617,86%	-654,32%	-661,79%	-534,00%	-482,73%	-550,31%	-562,26%
%I GUALA B&H	2,30%	1,60%	1,65%	0,70%	3,50%	6,70%	11,70%	14,90%
<b><math>s^2 = \sum_{i=1}^{i=n} \frac{(x_i - \bar{x})^2}{n - 1}</math></b>	0,00132640	0,00116100	0,00133346	0,00120856	0,00001852	0,00160970	0,00195171	0,00193306
Desviacion típica <b>S</b>	0,03641984	0,03407339	0,03651654	0,03476435	0,04303508	0,04012104	0,04417813	0,04396658
<b><math>n = \left(\frac{Z_{\alpha/2}S}{\bar{x}e}\right)^2</math></b>	14.842	150.872	54.883	215.836	603	807	928	888
<b><math>c_1 = \bar{x} - Z_{\alpha/2} \frac{s}{\sqrt{n}}</math></b>	-0,01331485	-0,00493206	-0,00771061	-0,00445693	-0,07055913	-0,05713041	-0,05878394	-0,05977049
<b><math>c_2 = \bar{x} + Z_{\alpha/2} \frac{s}{\sqrt{n}}</math></b>	-0,01012251	-0,00194539	-0,00450980	-0,00140970	-0,06678694	-0,05361364	-0,05491155	-0,05591665
COMPARACION	GPE/GPE-PS	GPE/GPE-ADF	GPE-ADF-PS	GPE/GP	GPE/GP-PS	GPE/GP-ADF	GPE/GP-PS-ADF	
<b><math>\bar{x} = \bar{x}_1 - \bar{x}_2</math></b>	0,00827996	0,00560848	0,00878537	-0,05695436	-0,04365334	-0,04512906	-0,05440485	
<b><math>s_x = \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}</math></b>	0,00003942	0,00004206	0,00004012	0,00002966	0,00004664	0,00005277	0,00005042	
<b><math>\bar{x} - Z_{\alpha/2}S_x</math></b>	0,00820270	0,00552604	0,00870672	-0,05701250	-0,04374476	-0,04523248	-0,05450368	
<b><math>\bar{x} + Z_{\alpha/2}S_x</math></b>	0,00835721	0,00569091	0,00886401	-0,05689622	-0,04356193	-0,04502564	-0,05430603	

Figura C.17: Resultados 2011. Funciones ampliadas (Entrenamiento 2008-10), 3.000 individuos.

ESTADÍSTICAS	ROI GE/GP TRADING RULES DATA SET 2012-2014, TEST 2015, 3000 INDIVIDUOS, 50 GENERACIONES, FUNCTION SET EXTENDIDO							
	GE	GE-PS	GE-ADF	GE-ADF-PS	GP	GP-PS	GP-ADF	GP-ADF-PS
$n$	2.000	2.000	2.000	2.000	2.000	2.000	2.000	2.000
$1-\alpha$	0,95	0,95	0,95	0,95	0,95	0,95	0,95	0,95
$z_{\alpha/2}$	1,96	1,96	1,96	1,96	1,96	1,96	1,96	1,96
$\epsilon$ MAXIMO	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05
$e = \frac{z_{\alpha/2}S}{\bar{x}\sqrt{n}}$	0,02434841	0,03526462	0,03446027	0,03088067	0,02910928	0,03101744	0,02910928	0,02101064
COMPLEJIDAD MEDIA	13,00	10,85	12,16	11,34	35,37	29,67	35,37	31,36
PROFUNDIDAD MEDIA	4,62	4,17	4,63	3,70	8,46	6,48	8,46	8,12
$\bar{x} = \frac{1}{n} \sum_{i=1}^{i=n} x_i$	-0,01604825	-0,01699954	-0,01927500	-0,01935347	-0,01490370	-0,01812581	-0,01490370	-0,01437625
% VECES MEJORA B&H Train.	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%
%MEJORA MEDIA Train.	19,82%	16,42%	13,41%	8,11%	32,89%	33,47%	32,89%	31,09%
%EMPEORA MEDIA Train.	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
% VECES MEJORA B&H Val.	1,85%	1,05%	2,05%	1,20%	1,35%	1,25%	1,35%	0,80%
%MEJORA MEDIA Val.	99,99%	110,14%	98,27%	101,89%	115,57%	117,02%	115,57%	101,83%
%EMPEORA MEDIA Val.	-466,08%	-538,67%	-514,49%	-626,45%	-462,55%	-462,49%	-462,55%	-413,39%
%IGUALA B&H	84,70%	85,55%	79,10%	83,30%	87,65%	80,60%	87,65%	88,40%
$s^2 = \sum_{i=1}^{i=n} \frac{(x_i - \bar{x})^2}{n-1}$	0,00007949	0,00018710	0,00022969	0,00018596	0,00000098	0,00016456	0,00009799	0,00004750
DESVIACIÓN TÍPICA $s$	0,00891574	0,01367840	0,01515556	0,01363655	0,00989884	0,01282810	0,00989884	0,00689198
$n = \left(\frac{z_{\alpha/2}S}{\bar{x}e}\right)^2$	474	995	950	763	678	770	678	353
$c_1 = \bar{x} - z_{\alpha/2} \frac{s}{\sqrt{n}}$	-0,01643900	-0,01759903	-0,01993922	-0,01995112	-0,01533753	-0,01868803	-0,01533753	-0,01467830
$c_2 = \bar{x} + z_{\alpha/2} \frac{s}{\sqrt{n}}$	-0,01565751	-0,01640006	-0,01861078	-0,01875582	-0,01446986	-0,01756360	-0,01446986	-0,01407419
COMPARACION	GPE/GPE-PS	GPE/GPE-ADF	GPE-ADF-PS	GPE/GP	GPE/GP-PS	GPE/GP-ADF	GPE/GP-PS-ADF	
$\bar{x} = \bar{x}_1 - \bar{x}_2$	0,00095129	0,00322675	0,00330522	-0,00114456	0,00207756	-0,00114456	-0,00262330	
$s_x = \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}$	0,00000455	0,00000543	0,00000452	0,00000178	0,00000409	0,00000282	0,00000432	
$\bar{x} - z_{\alpha/2} s_x$	0,00094238	0,00321609	0,00329635	-0,00114804	0,00206955	-0,00115009	-0,00263176	
$\bar{x} + z_{\alpha/2} s_x$	0,00096020	0,00323740	0,00331408	-0,00114108	0,00208557	-0,00113903	-0,00261484	

Figura C.18: Resultados 2015. Funciones ampliadas (Entrenamiento 2012-14), 3.000 individuos.

## C.4.2 Estadísticos

Estadísticas dataset 2004-2006 test 2007 3000-50 FS extendido test					
	Media	Mediana	Var.	Max	Min
ge	0,054955784	0,042128168	0,00145325	0,1625893	-0,06493377
geAd	0,049671885	0,042128168	0,001001555	0,18207785	-0,10659383
geAdfPa	0,048793784	0,042128168	0,000848127	0,1625893	-0,041227724
gePa	0,05375038	0,042128168	0,001333424	0,1514498	-0,07151019
gp	0,059927952	0,058848176	0,001449367	0,16179425	-0,08527455
gpAdf	0,060237776	0,058868606	0,001518581	0,19003828	-0,087709725
gpAdfPa	0,060189792	0,058868606	0,001501962	0,19003828	-0,07427338
gpPa	0,057035219	0,05684242	0,001353997	0,1651472	-0,07252713
Estadísticas dataset 2008-2010 test 2011 3000-50 extendido test					
	Media	Mediana	Var.	Max	Min
ge	-0,01171868	0,025042504	0,004821398	0,15225191	-0,33085275
geAdf	-0,006110205	0,025042504	0,004117952	0,13570783	-0,3031043
geAdfPa	-0,002933315	0,025042504	0,003820542	0,19692053	-0,2792663
gePa	-0,003438723	0,025042504	0,003743186	0,13570783	-0,33085275
gp	-0,068673037	-0,065565616	0,006311294	0,1946498	-0,38249722
gpAdf	-0,056847743	-0,041951198	0,006027827	0,16957054	-0,36780655
gpAdfPa	-0,057843574	-0,042712713	0,006229812	0,17422238	-0,34062696
gpPa	-0,055372024	-0,048901156	0,005604613	0,16957054	-0,340685
Estadísticas dataset 2012-2014 test 2015 3000-50 extendido test					
	Media	Mediana	Var.	Max	Min
ge	-0,016048255	-0,010121487	0,00042889	0,047253992	-0,21810734
geAdf	-0,019275001	-0,010121487	7,21E-04	0,07247244	-0,26471087
geAdfPa	-0,019353472	-0,010121487	7,92E-04	0,016278539	-0,16033211
gePa	-0,016999543	-0,010121487	5,65E-04	0,02252869	-0,17054616
gp	-0,017988185	-0,010121487	0,0005878	0,023488695	-0,26471087
gpAdf	-0,014903695	-0,010121487	0,000357429	0,028885301	-0,20216174
gpAdfPa	-0,014376247	-0,010121487	0,000297015	0,02252869	-0,22835211
gpPa	-0,018125812	-0,010121487	0,000528372	0,053672556	-0,17804097

Figura C.19: Estadísticas configuración extendida 3.000 individuos.

Notas:

2004-2006 Rbh 0,0310248290.

2008-2010 Rbh -0,0171277130.

2012-2014 Rbh -0,0101214870.

(ge) Gramática evolutiva.

(gp) Programación genética.

(Adf) Funciones automáticamente definidas.

(Pa) Control de complejidad mediante parsimonia.

## C.4.3 Contrastes estadísticos

Contrastes dataset 2004-2006 test 2007 3000-50 extendido validación								
	ge	geAd	geAdfPa	gePa	gp	gpAdf	gpAdfPa	gpPa
ge	NA	NA	NA	NA	NA	NA	NA	NA
geAd	=	NA	NA	NA	NA	NA	NA	NA
geAdfPa	=	=	NA	NA	NA	NA	NA	NA
gePa	=	++	++	NA	NA	NA	NA	NA
gp	++	++	++	++	NA	NA	NA	NA
gpAdf	++	++	++	++	=	NA	NA	NA
gpAdfPa	+	++	++	++	=	=	NA	NA
gpPa	=	++	++	=	--	--	--	NA
Contrastes dataset 2008-2010 test 2011 3000-50 extendido validación								
	ge	geAdf	geAdfPa	gePa	gp	gpAdf	gpAdfPa	gpPa
ge	NA	NA	NA	NA	NA	NA	NA	NA
geAdf	++	NA	NA	NA	NA	NA	NA	NA
geAdfPa	++	++	NA	NA	NA	NA	NA	NA
gePa	++	=	=	NA	NA	NA	NA	NA
gp	--	--	--	--	NA	NA	NA	NA
gpAdf	--	--	--	--	++	NA	NA	NA
gpAdfPa	--	--	--	--	++	=	NA	NA
gpPa	--	--	--	--	++	=	=	NA
Contrastes dataset 2012-2014 test 2015 3000-50 extendido validación								
	ge	geAdf	geAdfPa	gePa	gp	gpAdf	gpAdfPa	gpPa
ge	NA	NA	NA	NA	NA	NA	NA	NA
geAdf	=	NA	NA	NA	NA	NA	NA	NA
geAdfPa	=	+	NA	NA	NA	NA	NA	NA
gePa	=	++	+	NA	NA	NA	NA	NA
gp	=	=	=	=	NA	NA	NA	NA
gpAdf	+	++	++	++	++	NA	NA	NA
gpAdfPa	+	++	++	++	++	=	NA	NA
gpPa	=	=	=	=	=	=	=	NA

Figura C.20: Contrastes conjunto de funciones ampliado 3.000 individuos.

Notas:

(ge) Gramática evolutiva.

(gp) Programación genética.

(Adf) Funciones automáticamente definidas.

(Pa) Control de complejidad mediante parsimonia.

### C.5 Coste computacional

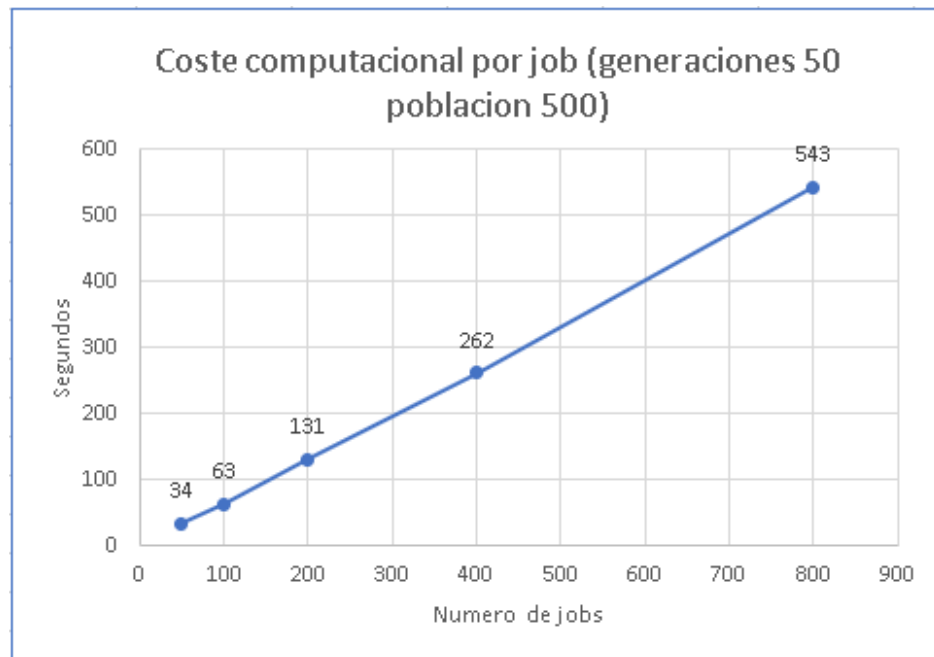


Figura C.21: Coste computacional en función del número de simulaciones.

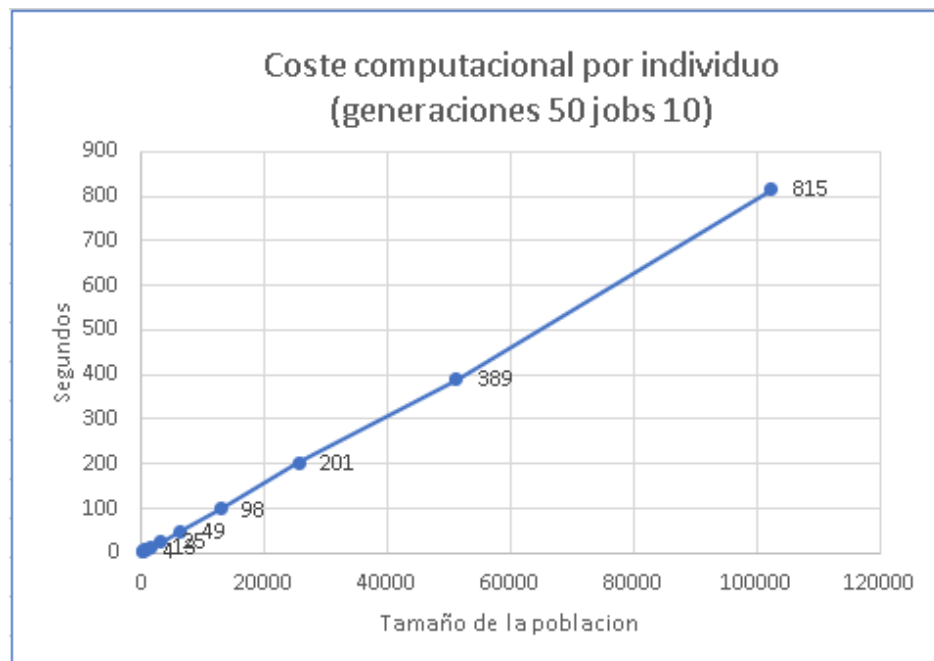
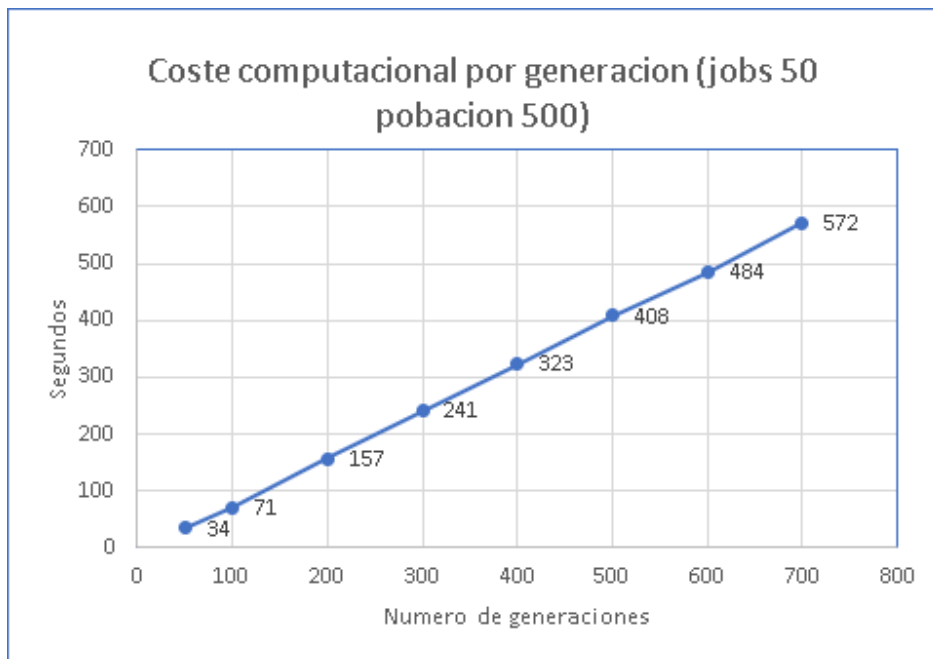


Figura C.22: Coste computacional en función del número de individuos.



**Figura C.23:** Coste computacional en función del número de generaciones.



# Glosario

## **Alelo**

Cada una de las posibles formas o valores que puede adoptar el gen de un cromosoma. 53, 93

## **Algoritmos evolutivos**

Conjunto de implementaciones concretas de la técnica de computación evolutiva, que incluyen entre otros los siguientes paradigmas: programación evolutiva, estrategias evolutivas, algoritmos genéticos y programación genética. 2

## **Algoritmos genéticos**

En ciencia de la computación e investigación operativa, un algoritmo genético (AG) es una técnica metaheurística, inspirada en el proceso de selección natural, que pertenece a una clase más amplia de algoritmos evolutivos (AE). Los algoritmos genéticos se usan comúnmente para generar soluciones a problemas de optimización, haciendo evolucionar una población de individuos sometiéndola a acciones aleatorias, semejantes a las que actúan en la evolución biológica (cruce, mutación, selección). 2

## **Application Program Interface**

Constituye un interfaz de programación que proporciona un conjunto de subrutinas, funciones, métodos y procedimientos, reunidos en una librería de desarrollo, que puede ser utilizado por otro software dentro de una capa de abstracción. 100

**Backus-Naur Formalism**

La forma Backus-Naur es un metalenguaje, o técnica de notación para definir gramáticas libres de contexto, que se usa a menudo para describir la sintaxis de los lenguajes utilizados en computación, tales como lenguajes de programación. 90

**Bolsa de Valores do Estado de São Paulo**

El Índice Bovespa es un índice bursátil, compuesto de unas 50 compañías que cotizan en la Bolsa de Sao Paulo (Bovespa: Bolsa de Valores do Estado de São PAulo). 28

**Buy and Hold**

En la estrategia de inversión comprar y mantener, un inversor compra valores bursátiles y los mantiene durante un largo periodo, con el objetivo de que su valor se incremente gradualmente durante el período de tiempo que pase hasta que se vendan. Se basa en la hipótesis de que, a la larga, los mercados financieros ofrecen una buena tasa de rentabilidad, incluso en periodos de volatilidad excesiva. 7

**Cartesian Genetic Programming**

La programación genética cartesiana, es una forma de programación genética que utiliza una representación gráfica para codificar programas informáticos. Se llama 'cartesiana' porque representa un programa mediante la utilización de una cuadrícula bidimensional de nodos. 2

**Computación evolutiva**

La computación evolutiva forma un subcampo de la IA, que comprende la familia de algoritmos de optimización global inspirados en la evolución biológica. En términos técnicos, son una familia de solucionadores de problemas basados en prueba y error, que proporcionan una optimización de carácter metaheurística o estocástica, utilizando conceptos de la evolución y selección natural de Charles Darwin [48] sobre una población. 1

**Deep Neural Network**

Es una implementación específica de una técnica de aprendizaje profundo utilizando redes neuronales. En algunos casos, una red neuronal profunda

puede ser una red neuronal convolucional, aunque, según algunas definiciones, una red neuronal profunda puede ser una red neuronal con muchas capas ocultas, incluso cuando no se realiza la convolución. 8

### **Ephemeral Random Constants**

Constantes generadas aleatoriamente y que a partir de ese momento permanecen invariables durante la ejecución del algoritmo. 20

### **Estrategia evolutiva**

En informática, una estrategia evolutiva (ES) es una técnica de optimización basada en las ideas de la evolución biológica. Pertenece a la clase general de computación evolutiva (CE) o metodologías de evolución artificial. 10

### **Extended Compact Genetic Programming**

El algoritmo genético compacto extendido, propuesto por Harik [87], se basa en la idea clave de que la elección de una buena distribución de probabilidad es equivalente al aprendizaje por vinculación. La medida de una buena distribución se cuantifica en función de los modelos de longitud de descripción mínima (MDL). El concepto fundamental detrás de los modelos MDL es que, suponiendo que todas las cosas son iguales, las distribuciones más simples son mejores que las complejas. La restricción MDL penaliza los modelos imprecisos y complejos, lo que lleva a una distribución de probabilidad óptima. Por lo tanto, la restricción MDL reformula el problema de encontrar una buena distribución como un problema de optimización que minimiza tanto el modelo de probabilidad como la representación de la población. 2

### **Fenotipo**

Solución candidata en un algoritmo evolutivo, después de ser descodificado a partir del genotipo. 41, 43, 80, 89

### **Fitness**

Métrica de calidad de un individuo para optimizar en un algoritmo evolutivo. En castellano se la suele conocer como función de aptitud, función de ajuste, o función de supervivencia. 10, 42

### **Full**

Algoritmo de inicialización ideado por Koza. 46, 47, 50

**Funciones definidas automáticamente**

Las funciones definidas automáticamente, fueron introducidas por primera vez por Koza como una forma de reutilizar el código en la programación genética. 9, 70

**Genotipo**

Representación computacional de una solución candidata en un algoritmo evolutivo, de acuerdo con una cierta codificación, sobre la cual se aplican los operadores evolutivos. 41, 43, 51, 63, 65, 80, 81, 85, 89, 114

**Gramáticas evolutivas**

Técnica de computación evolutiva en la que los individuos se codifican mediante cadenas de enteros, asociados a un lenguaje que puede ser generado por una gramática específica, de modo que el algoritmo aplicará operadores genéticos como la selección, recombinación o mutación, evolucionando estas cadenas y esperando que la calidad de los individuos mejore a medida que se suceden las generaciones. 2

**Grow**

Algoritmo de inicialización ideado por Koza. 46–48, 50

**High Frequency Trading**

En los mercados financieros, la negociación de alta frecuencia (HFT) es un tipo de negociación algorítmica caracterizada por altas velocidades de operación, altas tasas de rotación de activos y altos ratios de ordenes/operaciones que aprovechan datos financieros de alta frecuencia mediante herramientas de negociación electrónica. [5] 3

**Hill-Climbing**

En ciencia de la computación, el algoritmo "Hill Climbing" es una técnica de optimización matemática que pertenece a la familia de algoritmos basados en búsqueda local. También recibe el nombre de "Algoritmo de Escalada Simple" o "Ascenso de colinas". Su proceso iterativo se basa en remplazar la solución actual, con otra candidata generada al azar en el entorno cercano, variando un único elemento de la solución. Si se produce una mejora el candidato se

convierte en la solución actual. El proceso termina si no existe mejora posible, o si expira el tiempo de ejecución asignado. 2

### **Hipótesis del mercado eficiente**

La hipótesis del mercado eficiente (EMH) es una teoría de la economía financiera, que establece que los precios de los activos reflejan completamente toda la información disponible. Una implicación directa es que es imposible "ganarle al mercado" de forma sistemática y ajustada al riesgo ya que los precios del mercado solo deben reaccionar a la nueva información. 4

### **Indice Bursátil Chino Hang Seng**

Hang Seng Index (abreviado: HSI) es el principal índice bursátil chino de Hong Kong. 25

### **Indice de fuerza relativa**

El índice de fuerza relativa (Relative Strength Index o RSI, en inglés) es un indicador técnico utilizado en el análisis de los mercados financieros desarrollado por Wilder J. (1978) [215]. Está destinado a registrar la fortaleza o debilidad presente e histórica de una acción o del mercado en función de los precios de cierre de sesión en un período de negociación cercano. El indicador no debe confundirse con la medida de fuerza relativa. El RSI es un oscilador clasificado como de momento, que mide la velocidad y la magnitud de los movimientos y oscilaciones de precios direccionales. El impulso es la tasa de la subida o caída en el precio. El RSI calcula el ímpetu como una medida de la relación entre el cierre de sesión de mayor precio y el cierre inferior: las acciones que han tenido cambios positivos más frecuentes o más fuertes tienen un RSI mayor que las acciones que han tenido cambios negativos más frecuentes o más fuertes. 37

### **Indice de la Bolsa de valores de Alemania**

El DAX, DAX 30 o DAX Xetra (en alemán: Deutscher Aktienindex) es el índice de acciones de las 30 compañías más importantes de Alemania, que cotizan en la Bolsa de Frankfurt. 25

### **Indice de la bolsa de valores española de Madrid**

El índice IBEX35 es un índice bursátil compuesto, de las 35 compañías de mayor capitalización que cotizan en la bolsa española de Madrid. 33

**Input Output Operations Per Second**

Las operaciones de entrada/salida por segundo (IOPS) constituyen una medición de rendimiento de entrada/salida que se utiliza para caracterizar dispositivos de almacenamiento de computadora como unidades de disco duro (HDD), unidades de estado sólido (SSD) y redes de área de almacenamiento (SAN). Al igual que los puntos de referencia (benchmarks), los números de IOPS publicados por los fabricantes de dispositivos de almacenamiento no se relacionan directamente con el rendimiento real de las aplicaciones. 101

**Integrated Development Environment**

Un entorno de desarrollo integrado de software (en inglés: Integrated Development Environment), es una aplicación informática que proporciona servicios de desarrollo de software (edición, compilación, depuración, control de versiones etc.) que facilitan al programador las labores de programación. 100

**Inteligencia artificial**

Área de conocimiento, dentro de la Ingeniería Informática, dirigida al desarrollo de software y hardware que presenta cierto comportamiento inteligente, entendiendo como tal el que normalmente está presente y así se reconoce en la inteligencia humana. 1

**Iterated Local Search**

La búsqueda local es un método de tipo heurístico empleado para resolver problemas de optimización de complejidad computacional difíciles. La búsqueda local se puede utilizar en problemas de maximización, es decir, aquellos problemas que sea posible formularlos como una búsqueda para encontrar una solución que maximice un criterio, entre varias soluciones candidatas. Los algoritmos de búsqueda local se mueven de una solución candidata a otra, en el espacio de búsqueda, mediante la aplicación de cambios pequeños cambios locales, hasta que, o bien se encuentra una solución que se considera óptima, o ha transcurrido un tiempo máximo determinado. 2

**Linear Genetic Programming**

La programación genética lineal (LGP) es un subconjunto particular de la programación genética, en la que los programas de computadora, en una población, se representan como una secuencia de instrucciones de un lenguaje

de programación imperativo o en lenguaje de máquina. El flujo de datos, basado en el grafo que resulta de un uso múltiple de los contenidos del registro, y la existencia de un código estructuralmente no efectivo (intrones), son dos diferencias principales de esta representación genética de la variante más común de programación genética basada en árboles (TGP). 2

### **Macros definidas automáticamente**

Las macros definidas automáticamente, son parecidas a las funciones definidas automáticamente en cuanto a que constituyen una forma de reutilizar el código en la programación genética, pero se diferencia de estas en la forma en la que se evalúan sus argumentos. 71

### **Mersenne Twister pseudorandom number generator PRNG**

El Mersenne Twister es un generador de números pseudoaleatorios desarrollado en 1997 por Makoto Matsumoto y Takuji Nishimura. Su nombre hace referencia a que la longitud de su periodo se corresponde con un número primo de Mersenne. La variante más reciente y más utilizada es el Mersenne Twister MT19937. 115

### **Metaheurística**

Técnica de optimización que está diseñada para funcionar de manera eficiente en grandes espacios de búsqueda, sin utilizar conocimientos específicos sobre el dominio, sino más bien con alguna métrica de calidad de soluciones candidatas. 1, 41

### **Multi Expression Programming**

La programación multiexpresión (MEP), es una variante de la programación genética que utiliza una representación lineal de los cromosomas. Los individuos en MEP son cadenas de genes, que codifican programas informáticos complejos. Cuando los individuos en MEP codifican expresiones, su representación es similar, a la forma en que los compiladores traducen las expresiones de C o Pascal en código de máquina [4] 2

### **Multiobjective Genetic Programming**

Adaptación de las técnicas de optimización multiobjetivo que buscan encontrar soluciones que sean óptimas, o al menos aceptables, según todos los

criterios de optimización deseados a la vez. En la mayoría de los casos, cambiar un algoritmo de PG de objetivo único a multiobjetivo requiere alguna alteración en la forma en que se realiza la selección. 2

### **Multitree Genetic Programming**

A diferencia de la representación convencional de PG donde cada individuo es representado por un solo árbol, cada individuo en MTGP está representado por un conjunto de árboles. 2

### **New York Stock Exchange**

La Bolsa de Nueva York (en inglés: New York Stock Exchange, NYSE), es el mayor mercado de valores bursátiles del mundo, tanto en volumen monetario como en el número de empresas adscritas. 23

### **No Free Lunch**

En complejidad computacional y optimización, el teorema "No hay almuerzo gratis" (No Free Lunch) establece que, para ciertos tipos de problemas matemáticos, el coste computacional de encontrar una solución, promediada sobre todos los problemas de esa clase, es el mismo para cualquier método de solución. Ninguna solución, por lo tanto, ofrece un "atajo". En computación, hay circunstancias en las que los resultados de todos los procedimientos, para resolver un tipo particular de problema, son estadísticamente idénticos. 6

### **Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization**

El NSGAI, cuyo nombre completo es (Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization and Fast Elitist), es un algoritmo evolutivo multiobjetivo y elitista, que se basa en un enfoque de clasificación de soluciones no dominantes del frente Pareto. Constituye una evolución del NSGA computacionalmente más rápido. 36

### **Particle Swarm Optimization**

La optimización por nube de partículas u optimización por enjambre de partículas hace referencia a una metaheurística que evoca el comportamiento de las partículas, enjambres, colonias de insectos, bandadas de aves, etc., en la naturaleza. 100



**Probabilistic Incremental Program Evolution**

La evolución probabilística incremental del programa, es una técnica para la generación automática de programas informáticos. Combina la probabilidad de codificación en un vector de las instrucciones de un lenguaje, el aprendizaje incremental basado en la población y los programas codificados en árboles, como los que se usan en algunas variantes de la programación genética (PG). PIPE genera iterativamente poblaciones sucesivas de programas funcionales de acuerdo con una distribución de probabilidad adaptativa sobre todos los programas posibles. Cada iteración utiliza el mejor programa para refinar la distribución. 2

**Probabilistic Tree-Creation 1**

Algoritmo de inicialización ideado por PTC1 (Probabilistic Tree-Creation 1) diseñado por Sean Luke como una modificación del algoritmo Grow. 48

**Producto interior bruto**

El producto interior bruto (PIB), es un indicador económico que refleja la producción total de bienes y servicios asociada a un país durante un determinado periodo de tiempo, normalmente un año. Este indicador se emplea a nivel internacional para valorar la actividad económica o riqueza de cada país. 5

**Programación genética**

En inteligencia artificial, la programación genética (PG) es una técnica metaheurística mediante la cual los programas de computadora se codifican como un conjunto de genes que luego se evolucionan utilizando un algoritmo evolutivo, y en donde el espacio de soluciones consiste en programas de ordenador. Los resultados son programas que pueden funcionar bien para solucionar una tarea predefinida. 2

**Random Sampled Fitness Genetic Programming**

El algoritmo central es básicamente como el estándar de PG, pero en lugar de calcular la aptitud del individuo sobre el conjunto completo de datos, se basa en segmentos seleccionados al azar con el objetivo de aumentar la robustez.

**Randombranch**

Randombranch, es el nombre dado al algoritmo de inicialización que toma un tamaño de árbol solicitado y garantiza un árbol de ese tamaño o "algo más pequeño". 46, 48, 49

**RandomWalk**

Un paseo aleatorio es un objeto matemático, conocido como proceso estocástico o aleatorio, que describe una ruta que consiste en una sucesión de pasos aleatorios en algún espacio matemático. 6

**Rate of Change Indicator**

Indicador de tasa de cambio (ROC) define la tasa de cambio del precio. Es un indicador técnico de impulso que mide la variación porcentual del precio entre el precio actual y el precio de  $n$  períodos anteriores. 103

**Simulated Annealing**

Es una metaheurística utilizada para aproximar la optimización global a un problema que cuente con un gran espacio de búsqueda. A menudo se usa cuando el espacio de búsqueda es discreto (por ejemplo, todos los recorridos que visitan un conjunto determinado de ciudades). Es adecuado en aquellos problemas en los que encontrar un óptimo global aproximado sea más importante, que encontrar un óptimo local preciso cuando se cuenta con un período de tiempo fijo limitado. En esos casos Simulated Annealing puede ser preferible a otras alternativas como el descenso de gradiente. 2

**Stack-based GP**

En la programación genética basada en la pila, los programas representados en la población en evolución se expresan en un lenguaje de programación basado en pila. Los lenguajes específicos varían entre los sistemas, pero la mayoría son similares a FORTH en la medida en que los programas se componen de instrucciones que toman argumentos de las pilas de datos y devuelven los resultados en estas. En la familia de lenguajes Push, que fueron diseñados específicamente para la programación genética, se proporciona una pila por separado para cada tipo de datos. El propio código del programa puede manipularse en pilas de datos y ejecutarse posteriormente. 2

**Standard & Poor's 500**

El índice bursátil Standard & Poor's 500, también conocido como S&P 500, es uno de los índices bursátiles más importantes de Estados Unidos. Se lo considera el índice más representativo de la situación real del mercado, y toma como referencia la capitalización bursátil de las 500 empresas más grandes que poseen acciones que cotizan en las bolsas NYSE o NASDAQ. 19

**Stochastic Universal Sampling**

Es una variación del mecanismo de selección de ruleta, propuesto por Baker [15] en el que los individuos son elegidos al menos una vez. 56

**Strongly Typed Genetic Programming**

La programación genética fuertemente tipificada (STGP) es una versión mejorada de la programación genética que impone restricciones de tipo de datos y cuyo uso de funciones genéricas y tipos de datos genéricos lo hace más poderoso que otros enfoques para la aplicación de restricciones de tipo. 3

**Takagi-Sugeno fuzzy inference system**

En el campo de la inteligencia artificial, el modelo Neuro Fuzzy es un proceso por el cual se derivan conclusiones a partir de premisas. El término Neurodifuso se refiere a combinaciones de redes neuronales artificiales y lógica difusa. 36

**Tehran Stock Exchange**

El índice TSE es el índice del mercado de valores iraní que se compone de más de las 350 compañías de mayor valor bursátil de Irán. 25

**Toronto Stock Exchange index 300**

El índice TSE 300 era un índice del mercado de valores canadiense que rastreaba los precios de las 300 acciones de mayor valor bursátil que se negociaban en la Bolsa de Valores de Toronto. Desde el 1 de mayo de 2002, ha sido reemplazado por el Índice Compuesto S&P TSX. 23

**Tree-based Genetic Programming**

La programación genética basada en árboles (TGP) fue la primera aplicación de la programación genética. En la TGP los programas de ordenador se representan en estructuras de árbol que se evalúan recursivamente para producir

las expresiones multivariantes resultantes. La nomenclatura tradicional establece que un nodo de árbol (o solo nodo) es un operador [+ , - , \* , /] y un nodo terminal (u hoja) es una variable [a , b , c , d]. Lisp fue el primer lenguaje de programación aplicado a TGP ya que la estructura de este lenguaje coincide con la estructura de los árboles. 3

### **Uniform**

Uniform, es el nombre dado al algoritmo de inicialización exactamente uniforme de un árbol genético. 46, 48–50, 69, 76

### **Uniform Resources Locator**

Un localizador uniforme de recursos (URL) (en inglés: Uniform Resources Locator), denominado coloquialmente una dirección web, es una referencia a un recurso web que especifica su ubicación en una red informática y un mecanismo para recuperarlo. Una URL es un tipo específico de identificador uniforme de recursos (URI), aunque muchas personas usan los dos términos indistintamente. Las URL aparecen con mayor frecuencia para hacer referencia a páginas web (HTTP), pero también se utiliza para la transferencia de archivos (FTP), correo electrónico (mailto), acceso a la base de datos (JDBC) y muchas otras aplicaciones [43]. 8

### **Wavelet**

Una wavelet es una oscilación similar a una onda con una amplitud que comienza en cero, aumenta y luego vuelve a cero. Por lo general, se puede visualizar como una "oscilación breve" como la registrada por un sismógrafo o un monitor cardíaco. En general, las wavelets se diseñan intencionalmente para tener propiedades específicas que las hacen útiles para el procesamiento de señales. Usando una técnica de "inversión, cambio, multiplicación e integración" llamada convolución, las ondas pueden combinarse con partes conocidas de una señal dañada para extraer información de las partes desconocidas. 30

# Acrónimos

## **ADF**

Funciones definidas automáticamente 9, 35, 70, 71, 91, 99, 104, 105, 135, 136, 138, 142, 149, 151–153, 156, 157, 160, 161, 205–207, 215

## **ADM**

Macros definidas automáticamente 71

## **AE**

Algoritmos evolutivos 2, 208

## **AG**

Algoritmos genéticos 2, 9, 32, 37, 50, 52, 54, 66, 72–74, 80, 83, 94, 95, 100, 163

## **API**

Application Program Interface 100

## **B&H**

Buy and Hold 7, 11, 20–29, 32, 38, 136, 137, 139, 140, 142–144, 149–157, 160, 161, 206, 207

## **BBAS3**

Banco do Brasil SA 28

## **BNF**

Backus-Naur Formalism 81, 82, 90, 104

## **BOVA11**

ISHS IBOVESPA FDO indice ETF 28

**BOVESPA**

Bolsa de Valores do Estado de São Paulo 28

**CE**

Computación evolutiva 1–3, 7, 9, 11, 15, 17, 19, 34, 35, 43, 65, 71, 73, 79, 86, 99, 101, 102, 105, 108, 118, 133, 135, 136, 142, 160, 205, 208

**CGP**

Cartesian Genetic Programming 2

**CMIG4**

Companhia Energetica de Minas Gers, CEMIG Preference Shares 28

**CMS**

Recolector de basura de marca de barrido. (En inglés: Concurrent Mark Sweep (CMS) garbage collector) 111

**CNX**

CNX Nifty de Standard & Poor's 36

**DAX**

Indice de la Bolsa de valores de Alemania 25

**DDR4**

SDRAM (de las siglas en inglés, Double Data Rate type four Synchronous Dynamic Random-Access Memory) 101

**DNN**

Deep Neural Network 8

**DPX**

Double Point Crossover 86

**ECGP**

Extended Compact Genetic Programming 2

**ECJ**

Java Based Evolutionary Computation. 100, 101, 115

**ED**

Evolución diferencial 42

**EE**

Estrategia evolutiva 10, 54

**EMBR3**

IEmbreaer SA. 28

**EMH**

Hipótesis del mercado eficiente 5, 6, 22, 26

**ERC**

Ephemeral Random Constants 20, 31, 64

**ETF**

Exchange-Traded Fund 29

**FTP**

Protocolo de transferencia de ficheros (en inglés File Transfer Protocol o ftp)  
260

**GE**

Gramáticas evolutivas 2, 3, 7, 9–11, 15, 19, 26, 31–35, 41, 76, 79–82, 85, 86, 89–  
91, 94, 99–102, 104, 110, 111, 113–115, 117, 119, 133–136, 138, 140–143, 149–162,  
165, 166, 168, 170–172, 184, 186, 188, 189, 196, 199, 205–211

**GGBR4**

Gerdau SA Preference Shares 28

**HSI**

Indice Bursátil Chino Hang Seng 25

**HTF**

High Frequency Trading 26

**HTTP**

Protocolo de transferencia de hipertexto (en inglés: Hypertext Transfer Protocol o HTTP) 260

**IA**

Inteligencia artificial 1

**IBEX35**

Indice de la bolsa de valores española de Madrid 33

**IDE**

Integrated Development Environment 100

**IOPS**

Input Output Operations Per Second 101

**JDBC**

Conector de bases de datos en Java (en inglés: Java Database Connectivity) 260

**JVM**

Máquina virtual de Java (en inglés: Java Virtual Machine) 111

**LGP**

Linear Genetic Programming 2, 36

**LOWESS**

Locally Weighted Scatterplot Smoothing 38

**LR**

Linea de tendencia inferior 103

**LRH**

La hipótesis de razonamiento limitado 5



**MACD**

Convergencia divergencia de la media móvil 211

**MEP**

Multi Expression Programming 2, 36

**MOEA**

Multi-Objective Evolutionary Algorithms 36

**MOGP**

Multiobjective Genetic Programming 2

**MT**

Mersenne Twister pseudorandom number generator PRNG 116

**MTGP**

Multitree Genetic Programming 2

**MTH**

Marginal Trader Hypothesis 31

**NF**

Takagi-Sugeno fuzzy inference system 36

**NFL**

No Free Lunch 6

**NSGAI**

Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization  
36

**NSM**

Nasdaq Stock Market 36

**NYSE**

New York Stock Exchange 23

**PCT1**

Probabilistic Tree-Creation 1 48, 49

**PG**

Programación genética 2, 3, 6, 7, 9–11, 15, 19–31, 33, 34, 36, 38, 41, 42, 44, 46, 47, 50–53, 63–71, 73, 74, 76–81, 85, 86, 89–92, 94, 95, 99–102, 104, 108, 110–115, 117, 119, 133–136, 138, 140–143, 149–157, 159–162, 164, 171, 205–208, 211, 215

**PIB**

Producto interior bruto 5

**PIPE**

Probabilistic Incremental Program Evolution 2

**Postfix-GP**

Postfix Genetic Programming 2, 30

**PRI**

Indicador de resistencia de precios 103

**PRNG**

Pseudorandom number generator 255, 265

**PSO**

Particle Swarm Optimization 100

**RBH**

Es el retorno de la inversión obtenida comprando al inicio de un período y vendiendo al final 108

**RHH**

Ramped half-and-half es un algoritmo para la programación genética ideado por Koza 46–48, 50

**ROC**

Rate of Change Indicator 103

**RSFGP**

Random Sampled Fitness Genetic Programming 33

**RSI**

Indice de fuerza relativa 37

**S&P**

Standard & Poor's 500 19–24, 26, 29–32, 38

**SGP**

Stack-based GP 2

**SPX**

Single Point Crossover 86

**STGP**

Strongly Typed Genetic Programming 3, 30, 31, 69, 104, 113, 160, 206

**SUS**

Stochastic Universal Sampling 56

**TGP**

Tree-based Genetic Programming 3

**TLI**

Indicadores de línea de tendencia (en inglés: Trend Line Indicator 103

**TSE**

Tehran Stock Exchange 25

**TSE 300**

Toronto Stock Exchange index 300 23

**UPX**

Uniform Point Crossover 88

**UR**

Línea de tendencia superior 103

**URI**

Identificador uniforme de recursos (en inglés: Uniform Resources Identifier)  
260

**URL**

Uniform Resources Locator 8, 134, 171

**VALE5**

Indicador de resistencia de precios 28

# Bibliografía

- [1] ABRAHAM, A. Neuro fuzzy systems: State-of-the-art modeling techniques. In *International Work-Conference on Artificial Neural Networks* (2001), Springer, pp. 269–276.
- [2] AGAPITOS, A., BRABAZON, A., AND O’NEILL, M. Genetic programming with memory for financial trading. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (2016), vol. 9597, Springer, Cham, pp. 19–34.
- [3] AGUILAR-RIVERA, R., WITH . . . , M. V.-R. E. S., AND UNDEFINED 2015. Genetic algorithms and Darwinian approaches in financial applications: A survey. *Elsevier* (2015).
- [4] AHO, A. V., SETHI, R., AND ULLMAN, J. D. *Compilers: Principles, tools, and techniques*, 1986.
- [5] ALDRIDGE, I. *High-frequency trading: a practical guide to algorithmic strategies and trading systems*. wiley, 2013. ISBN 9781118343500.
- [6] ALEA RIERA, M. V., AND UNIVERSITAT DE BARCELONA. DEPARTAMENT D’ECONOMETRIA, E. I. E. E. *Estadística con SPSS v. 10.0*. Edicions Universitat de Barcelona, 2001. ISBN 9788483382578.
- [7] ALER, R., BORRAJO, D., AND ISASI, P. Using genetic programming to learn and improve control knowledge. *Artificial Intelligence* 141, 1-2 (2002), 29–56.
- [8] ALLEN, F., AND KARJALAINEN, R. Using genetic algorithms to find technical trading rules. *Journal of Financial Economics* 51, 2 (1999), 245–271.

- [9] ALONSO, AND SCHOTT, R. *Random generation of trees: random generators in computer science*. Kluwer Academic Publishers, 1995.
- [10] ALTENBERG, L. The schema theorem and price's theorem. In *Foundations of genetic algorithms*, vol. 3. Elsevier, 1995, pp. 23–49.
- [11] ANDERSON, T. W., AND DARLING, D. A. Asymptotic theory of certain “goodness of fit” criteria based on stochastic processes. *The annals of mathematical statistics* (1952), 193–212.
- [12] BABCOCK, B. *The Dow Jones-Irwin guide to trading systems*. Dow Jones-Irwin, 1989.
- [13] BABOVIC, V., AND KEIJZER, M. Genetic programming as a model induction engine. *Journal of Hydroinformatics* 2, 1 (2000), 35–60.
- [14] BAKER, J. E. Adaptive selection methods for genetic algorithms. In *Proceedings of an International Conference on Genetic Algorithms and their applications* (1985), Hillsdale, New Jersey, pp. 101–111.
- [15] BAKER, J. E. Reducing bias and inefficiency in the selection algorithm. In *Proceedings of the second international conference on genetic algorithms* (1987), vol. 206, pp. 14–21.
- [16] BALADA NICOLAU, F. *Farmacología y endocrinología del comportamiento*. Editorial UOC, 2012.
- [17] BECKER, L., AND SESHADRI, M. Comprehensibility and overfitting avoidance in genetic programming for technical trading rules. ... *Institute, Computer Science Technical Report* (2003), 1–87.
- [18] BECKER, L., AND SESHADRI, M. GP-evolved technical trading rules can outperform buy and hold. *Proceedings of the Sixth International Conference on Computational Intelligence and Natural Computing, Embassy Suites Hotel and Conference Center, Cary, North Carolina USA, September 26-30 2003* (2003).
- [19] BECVAR, J. *Robert R. Korfhage, Logic and Algorithms with Applications to the Computer and Information Sciences*. Wiley, 1971.

- [20] BERUTICH, J. M., LÓPEZ, F., LUNA, F., AND QUINTANA, D. Robust technical trading strategies using GP for algorithmic portfolio selection. *Expert Systems with Applications* 46 (2016), 307–315.
- [21] BIAIS, B., AND WOOLLEY, P. High frequency trading. *Manuscript, Toulouse University, IDEI* (2011).
- [22] BIDAU, C. J. Domestication through the Centuries: Darwin’s Ideas and Dmitry Belyaev’s Long-Term Experiment in Silver Foxes. *Gayana (Concepción)* 73 (2009), 55–72.
- [23] BISHOP, C. M., ET AL. *Neural networks for pattern recognition*. Oxford university press, 1995.
- [24] BLICKLE, T., AND THIELE, L. A comparison of selection schemes used in genetic algorithms, 1995.
- [25] BLICKLE, T., AND THIELE, L. A mathematical analysis of tournament selection. In *ICGA (1995)*, Citeseer, pp. 9–16.
- [26] BLICME, T., AND THIELE, L. A ComDarison of Selection Schemes Used iL Evolutionary Algorithms. *Evolutionary Computation*, 04 (1997), 361–3944.
- [27] BÖHM, W., AND GEYER-SCHULZ, A. Exact Uniform Initialization For Genetic Programming. *FOGA* (1996).
- [28] BOOKER, L. B., GOLDBERG, D. E., AND HOLLAND, J. H. Classifier systems and genetic algorithms. *Artificial Intelligence* 40, 1-3 (sep 1989), 235–282. doi: [https://doi.org/10.1016/0004-3702\(89\)90050-7](https://doi.org/10.1016/0004-3702(89)90050-7), ISBN 0004-3702, ISSN 00043702.
- [29] BRABAZON, A., AND O’NEILL, M. Evolving technical trading rules for spot foreign-exchange markets using grammatical evolution. *CMS* 1 (2004), 311–327.
- [30] BRABAZON, A., AND O’NEILL, M. *Biologically inspired algorithms for financial modelling*. Springer, 2006.
- [31] BRAMEIER, M., AND BANZHAF, W. *Linear Genetic Programming*. Springer, 2007. doi: <https://doi.org/10.1007/978-0-387-31030-5>, ISBN 978-0-387-31029-9, ISSN 1098-6596.

- [32] BRINDLE, A. Genetic algorithms for function optimization. *University of Alberta* (1981).
- [33] BROCK, W., LAKONISHOK, J., AND LEBARON, B. Simple Technical Trading Rules and The Stochastic Properties of Stock Returns, 1992.
- [34] BURJORJEE, K. The fundamental problem with the building block hypothesis. *arXiv preprint arXiv:0810.3356* (2008).
- [35] CASELLA, G., FIENBERG, S., AND OLKIN, I. *Elements in statistical learning*. Springer, Berlin, Heidelberg, 2005. doi: <https://doi.org/10.1007/978-1-4614-7138-7>, ISBN 978-1-4614-7137-0, ISSN 01665316.
- [36] CHABOUD, A. P., CHIQUOINE, B., HJALMARSSON, E., AND VEGA, C. Rise of the machines: algorithmic trading in the foreign exchange market. *The Journal of Finance* 69, 5 (2014), 2045–2084.
- [37] CHAKRABORTY, M., AND CHAKRABORTY, U. An analysis of linear ranking and binary tournament selection in genetic algorithms. *ICICS '97*, September (2002), 407–411.
- [38] CHELLAPILLA, K. Evolving computer programs without subtree crossover. *IEEE Transactions on Evolutionary Computation* 1, 3 (1997), 209–216.
- [39] CLEVELAND, W. S. Lowess: A program for smoothing scatterplots by robust locally weighted regression. *American Statistician* 35, 1 (1981), 54.
- [40] CONTRERAS, I., HIDALGO, J. I., AND NÚÑEZ-LETAMENDIA, L. A GA Combining Technical and Fundamental Analysis for Trading the Stock Market. In *A GA combining technical and fundamental analysis for trading the stock market*. Springer, Berlin, Heidelberg, 2012, pp. 174–183.
- [41] CONTRERAS, I., HIDALGO, J. I., AND NUNEZ-LETAMENDIA, L. Combining Technical Analysis and Grammatical Evolution in a Trading System. In *Applications of Evolutionary Computing, EvoApplications 2013: EvoCOMNET, EvoCOMPLEX, EvoENERGY, EvoFIN, EvoGAMES, EvoIASP, EvoINDUSTRY, EvoNUM, EvoPAR, EvoRISK, EvoROBOT, EvoSTOC*, vol. 7835. Springer, Berlin, Heidelberg, 2013, pp. 244–253.



- [42] COOK, R. D., AND HAWKINS, D. M. Unmasking multivariate outliers and leverage points: comment. *Journal of the American Statistical Association* 85, 411 (1990), 640–644.
- [43] COULOURIS, G. F., DOLLIMORE, J., AND KINDBERG, T. *Distributed systems: concepts and design*. pearson education, 2005.
- [44] CRAMER, N. L. A representation for the adaptive generation of simple sequential programs. In *Proceedings of an International Conference on Genetic Algorithms and the Applications* (Carnegie-Mellon University, Pittsburgh, PA, USA, 24-26 July 1985), J. J. Grefenstette, Ed., pp. 183–187.
- [45] DABHI, V. K., AND CHAUDHARY, S. Financial Time Series Modeling and Prediction Using Postfix-GP. *Computational Economics* 47, 2 (2016), 219–253. doi: <https://doi.org/10.1007/s10614-015-9482-y>, ISBN 9781479949106, ISSN 15729974.
- [46] DABHI, V. K., AND VIJ, S. K. Empirical modeling using symbolic regression via postfix genetic programming. In *Image Information Processing (ICIIP), 2011 International Conference on* (2011), IEEE, pp. 1–6.
- [47] DACOROGNA, M. M., GENÇAY, R., MÜLLER, U. A., AND PICTET, O. V. Effective return, risk aversion and drawdowns. *Physica A: Statistical Mechanics and its Applications* 289, 1-2 (2001), 229–248.
- [48] DARWIN, C. *On the origin of species, 1859*. Routledge, 2004.
- [49] DAWKINS, R. *The Selfish Gene: –with a new Introduction by the Author*. Oxford University Press, USA, 2006.
- [50] DE JONG, K. A. *An {Analysis} of the {Behavior} of a {Class} of {Genetic} {Adaptive} {Systems}*. PhD thesis, University of Michigan, 1975.
- [51] DEB, K., AGRAWAL, S., PRATAP, A., AND MEYARIVAN, T. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: Nsga-ii. In *International Conference on Parallel Problem Solving From Nature* (2000), Springer, pp. 849–858.
- [52] DEMPSEY, I., O’NEILL, M., AND BRABAZON, A. Live Trading with Grammatical Evolution. *GECCO 2004 Workshop Proceedings* (2004), 9137–9142. doi: <https://doi.org/10.1109/CEC.2006.1688631>, ISBN 0-7803-9487-9.

- [53] DEMPSTER, M. A. H., AND JONES, C. M. a Real Time Adaptive Trading System Using Genetic Programming. *QUANTITATIVE FINANCE VOLUME* (2001), 1–17.
- [54] DENG, S., AND SAKURAI, A. Foreign exchange trading rules using a single technical indicator from multiple timeframes. *Proceedings - 27th International Conference on Advanced Information Networking and Applications Workshops, WAINA 2013* (2013), 207–212. doi: <https://doi.org/10.1109/WAINA.2013.7>, ISBN 9780769549521.
- [55] ECJ. ECJ. <https://cs.gmu.edu/~eclab/projects/ecj/>. Accedido online en 2019-01-07.
- [56] ECLIPSE. Eclipse IDE for Java Developers | Eclipse Packages. <https://www.eclipse.org/downloads/packages/release/mars/r/eclipse-ide-java-developers>. Accedido online en 2019-01-07.
- [57] EIBEN, A. E., MICHALEWICZ, Z., SCHOENAUER, M., AND SMITH, J. E. Parameter control in evolutionary algorithms. *Studies in Computational Intelligence 54* (2007), 19–46.
- [58] EL PARLAMENTO EUROPEO Y EL CONSEJO DE LA UNIÓN EUROPEA. REGLAMENTO (UE) 2016/ 679 DEL PARLAMENTO EUROPEO Y DEL CONSEJO - de 27 de abril de 2016 - relativo a la protección de las personas físicas en lo que respecta al tratamiento de datos personales y a la libre circulación de estos datos y por el que se deroga la Directiva 95/ 46/ CE (Reglamento general de protección de datos). Tech. rep., El Parlamento Europeo y el Consejo de la Unión Europea, 2018. Accedido online en 2019-02-25 <https://eur-lex.europa.eu/legal-content/ES/TXT/PDF/?uri=CELEX:32016R0679&from=EN>.
- [59] ELDER, A. *Trading for a living: psychology, trading tactics, money management*, vol. 31. John Wiley & Sons, 1993.
- [60] ELLSON, J., GANSNER, E., KOUTSOFIOS, L., NORTH, S., WOODHULL, G., DESCRIPTION, S., AND TECHNOLOGIES, L. Graphviz — open source graph drawing tools. In *Lecture Notes in Computer Science* (2001), Springer-Verlag, pp. 483–484.

- [61] ESFAHANIPOUR, A., AND MOUSAVI, S. A genetic programming model to generate risk-adjusted technical trading rules in stock markets. *Expert Systems with Applications* 38, 7 (2011), 8438–8445.
- [62] FAMA, E. Efficient capital markets: A review of theory and empirical work. *Journal of Finance* 25 (1970), 383–417.
- [63] FAMA, E. F. The behavior of stock-market prices. *Journal of business* 38, 1 (1965), 34–105.
- [64] FAMA, E. F., WALKS, R., MARKET, S., AUTHOR, P., SOURCE, E. F. F., JOURNAL, F. A., AND FAMA, E. F. Random Walks in Stock Market Prices. *Financial Analysts Journal* 21, 5 (1965), 55–59.
- [65] FERREIRA, C. Automatically defined functions in gene expression programming. *Studies in Computational Intelligence* 13 (2006), 21–56.
- [66] FINGER, M., , COMPUTATION, R. W. J. O. L., AND UNDEFINED 2004. Approximate and limited reasoning: Semantics, proof theory, expressivity and control. *academic.oup.com* (2004).
- [67] FOGEL, D. B. Using Evolutionary Programming for Modeling: An Ocean Acoustic Example. *IEEE Journal of Oceanic Engineering* 17, 4 (1992), 333–340. doi: <https://doi.org/10.1109/48.180302>, ISBN 0364-9059, ISSN 15581691.
- [68] FORSYTHE, R., NELSON, F., NEUMANN, G. R., AND WRIGHT, J. Anatomy of an experimental political stock market. *The American Economic Review* (1992), 1142–1161.
- [69] FOUCAULT, T., HOMBERT, J., AND ROSU, I. News trading and speed. *The Journal of Finance* 71, 1 (2015), 335–382.
- [70] FRY, B. *Visualizing Data*, 1st ed. O'Reilly Media, 2008. ISBN 978-0-596-51455-6.
- [71] FYFE, C., MARNEY, J. P., AND TARBERT, H. Risk adjusted returns from technical trading: a genetic programming approach. *Applied Financial Economics* 15, 15 (2005), 1073–1077.

- [72] GABRIELSSON, P., JOHANSSON, U., AND KONIG, R. Co-evolving online high-frequency trading strategies using grammatical evolution. In *IEEE/IAFE Conference on Computational Intelligence for Financial Engineering, Proceedings (CIFEr)* (2014), pp. 473–480.
- [73] GAGNIUC, P. A. *Markov chains: from theory to implementation and experimentation*. John Wiley & Sons, 2017.
- [74] GARCÍA, S., QUINTANA, D., GALVÁN, I. M., AND ISASI, P. Multiobjective algorithms with resampling for portfolio optimization. *Computing and Informatics* 32, 4 (2013), 777–796.
- [75] GENNOTTE, G., AND JUNG, A. Investment strategies under transaction costs: the finite horizon case. *Management Science* 40, 3 (1994), 385–404.
- [76] GOETZ, B., PEIERLS, T., LEA, D., BLOCH, J., BOWBEER, J., AND HOLMES, D. *Java concurrency in practice*. Pearson Education, 2006.
- [77] GOLDBERG, D. E. *Genetic Algorithms in Search, Optimization, and Machine Learning*, vol. Addison-We. Addison Wesley, 1989.
- [78] GOLDBERG, D. E., AND DEB, K. A comparative analysis of selection schemes used in genetic algorithms. In *Foundations of genetic algorithms*, vol. 1. Elsevier, 1991, pp. 69–93.
- [79] GOLDBERG, D. E., DEB, K., AND CLARK, J. H. Genetic Algorithms, Noise, and the Sizing of Populations. *Complex Systems* 6 (1991), 333–362.
- [80] GREFENSTETTE, J. Optimization of Control Parameters for Genetic Algorithms. *Systems, Man and Cybernetics, IEEE Transactions on* 16 (1986), 122–128.
- [81] GRINSTEAD, C. M., AND SNELL, J. L. Central limit theorem. *Introduction to probability* (1997), 325–355.
- [82] GROSAN, C., AND ABRAHAM, A. Solving no free lunch issues from a practical perspective. In *Proceedings of Ninth International Conference on Cognitive and Neural Systems, ICCNS* (2005), vol. 5.
- [83] GROSAN, C., AND ABRAHAM, A. Stock market modeling using genetic programming ensembles. In *Genetic systems programming*. Springer, 2006, pp. 131–146.

- [84] GURARI, E. *Cis 680: Data structures: Chapter 19: Backtracking algorithms*, 1999.
- [85] GYPTEAU, J., OTERO, F. E. B., AND KAMPOURIDIS, M. Generating directional change based trading strategies with genetic programming. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (2015), vol. 9028, pp. 267–278.
- [86] HANSEN, L. K., AND SALAMON, P. Neural network ensembles. *IEEE transactions on pattern analysis and machine intelligence* 12, 10 (1990), 993–1001.
- [87] HARIK, G. R., LOBO, F. G., AND SASTRY, K. Linkage learning via probabilistic modeling in the extended compact genetic algorithm (ECGA). *Studies in Computational Intelligence* (2007). doi: [https://doi.org/10.1007/978-3-540-34954-9\\_3](https://doi.org/10.1007/978-3-540-34954-9_3), ISBN 3540349537, ISSN 1860949X.
- [88] HAYEK, F. A. Monetary theory and the trade cycle. *Journal of the Royal Statistical Society* 96, 2 (feb 1933), 319–321. doi: <https://doi.org/10.2307/2341796>, ISSN = 1469-7688.
- [89] HENDERSON, G. A. *El arte de elaborar el estado del arte en una investigación*. CIADEG-TEC 2014, 2014.
- [90] HINTERDING, R. Gaussian mutation and self-adaption for numeric genetic algorithms. In *Proceedings of 1995 IEEE International Conference on Evolutionary Computation* (1995), vol. 1, IEEE, p. 384. doi: <https://doi.org/10.1109/ICEC.1995.489178> ISBN 0-7803-2759-4, ISSN 1476-4679.
- [91] HINTON, G. E., AND NOWLAN, S. J. How learning can guide evolution. *Complex systems* 1, 3 (1987), 495–502.
- [92] HOLLAND, J. H. *Adaptation in Natural and Artificial Systems*, vol. Ann Arbor. The University of Michigan Press, 1975.
- [93] HOLLAND, J. H., ET AL. *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. MIT press, 1992.

- [94] HOLZINGER, K., PALADE, V., RABADAN, R., AND HOLZINGER, A. Interactive Knowledge Discovery and Data Mining in Biomedical Informatics. *LNCS 8401 8401*, November (2014), 35—56. doi: <https://doi.org/10.1007/978-3-662-43968-5>, ISBN 978-3-662-43967-8, ISSN 09622802.
- [95] HONGGUANG, L., AND PING, J. Generating Intraday Trading Rules on Index Future Markets Using Genetic Programming. *International Journal of Trade, Economics and Finance* 6, 2 (2015), 112–116.
- [96] HOW, J., LING, M., AND VERHOEVEN, P. Does size matter? A genetic programming approach to technical trading. *Quantitative Finance* 10, 2 (feb 2010), 131–140. ISSN 1469-7688.
- [97] IBA, H. Random tree generation for genetic programming. In *International Conference on Parallel Problem Solving from Nature* (1996), Springer, pp. 144–153.
- [98] INGRAM OLKIN, HAROLD HOTELLING, ET AL. H. Levene (1960) Robust tests for equality of variances. *Stanford University Press* (1960), 278–292.
- [99] JAIN, P. *Buffett beyond value : why Warren Buffett looks to growth and management when investing*. John Wiley & Sons, 2010. ISBN 0470608951.
- [100] JAMES, G., WITTEN, D., HASTIE, T., AND TIBSHIRANI, R. *An introduction to statistical learning*, vol. 112. Springer, 2013.
- [101] JANG, J.-S. R., SUN, C.-T., AND MIZUTANI, E. Neuro-fuzzy and soft computing; a computational approach to learning and machine intelligence. *Prentice Hall, Upper Saddle River* (1997).
- [102] JANSEN, S. *Testing market imperfections via genetic programming*. PhD thesis, Univertitat Hohenheim, 2011.
- [103] JAVA. java.com: Java y Tú. <https://www.java.com/es/>. Accedido online en 2019-01-08.
- [104] KENNEDY, J., AND EBERHART, R. Particle swarm optimization, proceedings of ieee international conference on neural networks (icnn'95) in, 1995.

- [105] KHACHATURYAN, A., SEMENOVSKAYA, S., AND VAINSTEIN, B. A statistical-thermodynamic approach to determination of structure amplitude phases. *Sov. Phys. Crystallography* 24, 5 (1979), 519–524.
- [106] KIM, Y., AND SOHN, S. Y. Stock fraud detection using peer group analysis. *Expert Systems with Applications* 39, 10 (2012), 8986–8992.
- [107] KIRKPATRICK, S., GELATT, C. D., AND VECCHI, M. P. Optimization by simulated annealing. *science* 220, 4598 (1983), 671–680.
- [108] KOLBADI, P., AND AHMADINIA, H. Examining sharp, sortino and sterling ratios in portfolio management, evidence from tehran stock exchange. *International Journal of Business and Management* 6, 4 (2011), 222.
- [109] KOLMOGOROV, A. Sulla determinazione empirica di una legge di distribuzione, giornale dell'istituto italiano degli attuari 4, 461 (1933). translated in an shiryayev. *Selected Works of AN Kolmogorov* 2 (1933), 139.
- [110] KOZA, J. R. Discovery of a main program and reusable subroutines using genetic programming. In *Proceedings of the Fifth Workshop on Neural Networks: An International Conference on Computational Intelligence: Neural Networks, Fuzzy Systems, Evolutionary Programming, and Virtual Reality* (1993), pp. 109–118.
- [111] KOZA, J. R. Genetic Programming 2: Automatic Discovery of Reusable Programs. *Artificial Life* 1, 4 (1994), 267–292.
- [112] KOZA, J. R., AND ANGELINE, P. J. *Genetic programming: On the programming of computers by means of natural selection*, vol. 33. MIT press, 1992.
- [113] LAKEMEYER, G. Limited reasoning in first-order knowledge bases. *Elsevier* (1994).
- [114] LANGDON, W. B., AND POLI, R. Fitness causes bloat: Mutation. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (1998), vol. 1391, pp. 37–48.
- [115] LANGDON, W. B., AND POLI, R. Introduction. In *Foundations of Genetic Programming*. Springer, 2002, pp. 1–16.

- [116] LANGDON, W. B., POLI, R., MCPHEE, N. F., AND KOZA, J. R. Genetic programming: An introduction and tutorial, with a survey of techniques and applications. *Studies in Computational Intelligence* 115 (2008), 927–1028.
- [117] LILLIEFORS, H. W. On the Kolmogorov-Smirnov-Test for normality with mean and variance unknown. *Journal of the American Statistical Association* 62, 318 (jun 1967), 399–402. doi: <https://doi.org/10.1080/01621459.1967.10482916>, ISBN 01621459, ISSN 01621459.
- [118] LO, A. W., MAMAYSKY, H., AND WANG, J. Foundations of technical analysis: computational algorithms, statistical inference, and empirical implementation. *The Journal of Finance* 55, 4 (2002), 1705–1765.
- [119] LOBO, F. G., LIMA, C. F., AND MICHALEWICZ, Z. *Parameter setting in evolutionary algorithms*. Springer, 2007.
- [120] LOHPETCH, D., AND CORNE, D. Discovering effective technical trading rules with genetic programming: Towards robustly outperforming buy-and-hold. In *2009 World Congress on Nature and Biologically Inspired Computing, NABIC 2009 - Proceedings* (2009), pp. 439–444.
- [121] LOHPETCH, D., AND CORNE, D. Outperforming buy-and-hold with evolved technical trading rules: Daily, weekly and monthly trading. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (2010), vol. 6025 LNCS, pp. 171–181.
- [122] LOURENÇO, H. R., MARTIN, O. C., AND STÜTZLE, T. Iterated local search: Framework and applications. In *Handbook of Metaheuristics*. Springer, 2019, pp. 129–168.
- [123] LUCAS, S. Structuring chromosomes for context-free grammar evolution. In *Evolutionary Computation, 1994. IEEE World Congress on Computational Intelligence., Proceedings of the First IEEE Conference on* (1994), IEEE, pp. 130–135.
- [124] LUENGO, S., WINKLER, S., BARRERO, D. F., AND CASTAÑO, B. Optimization of trading rules for the spanish stock market by genetic programming. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (2015), vol. 9101, Springer, Cham, pp. 623–634.



- [125] ŁUKASIEWICZ, J. *Aristotle's syllogistic from the standpoint of modern formal logic*. Oxford University Press, 1951.
- [126] LUKE, S. Code Growth Is Not Caused by Introns. In *In Whitley, D. (Ed.), Late Breaking Papers at the 2000 Genetic and Evolutionary Computation Conference (pp. 228– 235). Las Vegas (2000)*, pp. 228–235.
- [127] LUKE, S. Two fast tree-creation algorithms for genetic programming. *IEEE Transactions on Evolutionary Computation* 4, 3 (2000), 274–283.
- [128] LUKE, S. *Sean Luke: essentials of metaheuristics*, vol. 12. Lulu, 2011.
- [129] LUKE, S. *The ECJ Owner 's Manual*. George Mason University, 2013.
- [130] LUKE, S., AND PANAIT, L. A survey and comparison of tree generation algorithms. *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001)* (2001), 81–88.
- [131] LUKE, S., AND PANAIT, L. A comparison of bloat control methods for genetic programming. *Evolutionary Computation* 14, 3 (2006), 309–344.
- [132] MACHADO, J., NEVES, R., AND HORTA, N. Developing multi-time frame trading rules with a trend following strategy, using GA. In *17th Genetic and Evolutionary Computation Conference, GECCO 2015* (2015), pp. 765–766. doi: <https://doi.org/10.1145/2739482.2764885>, ISBN 9781450334884.
- [133] MANAHOV, V., HUDSON, R., AND HOQUE, H. Return predictability and the 'wisdom of crowds': Genetic Programming trading algorithms, the Marginal Trader Hypothesis and the Hayek Hypothesis. *Journal of International Financial Markets, Institutions and Money* 37 (2015), 85–98.
- [134] MARCOS; RIVERO GESTAL (DANIEL; RABUÑAL, JUAN RAMÓN; DORADO, J. P. A., AND GESTAL, M. *Introducción a los algoritmos genéticos y la programación genética*. Universidade da Coruña, 2010.
- [135] MARTÍN, C., QUINTANA, D., AND ISASI, P. Evolution of trading strategies with flexible structures: A configuration comparison. *Neurocomputing* (nov 2018). doi: <https://10.1016/J.NEUCOM.2018.11.062>, ISSN 0925-2312.

- [136] MATSUMOTO, M., AND NISHIMURA, T. Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. *ACM Transactions on Modeling and Computer Simulation (TOMACS)* 8, 1 (1998), 3–30.
- [137] MCPHEE, N. F., AND MILLER, J. D. Accurate Replication in Genetic Programming. *Genetic Algorithms: Proceedings of the Sixth International Conference (ICGA95)* (1995), 303–309.
- [138] MICHALEWICZ, Z. *Genetic Algorithms + Data Structures = Evolution Programs*, 1996.
- [139] MILLÁN, D. B., BOTICARIO, J. G., AND VIÑUELA, P. I. *Aprendizaje automático*. Sanz y Torres, 2006.
- [140] MILLER, J. F., Ed. *Cartesian Genetic Programming*. Natural Computing Series. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011. doi: <https://doi.org/10.1007/978-3-642-17310-3> ISBN 978-3-642-17309-7.
- [141] MONTANA, D. J. Strongly Typed Genetic Programming. *Evolutionary Computation* 3, 2 (1995), 199–230.
- [142] MOUSAVI, S., ESFAHANIPOUR, A., AND ZARANDI, M. H. F. A novel approach to dynamic portfolio trading system using multitree genetic programming. *Knowledge-Based Systems* 66 (2014), 68–81.
- [143] MÜHLENBEIN, H., AND SCHLIERKAMP-VOOSEN, D. Predictive models for the breeder genetic algorithm i. continuous parameter optimization. *Evolutionary computation* 1, 1 (1993), 25–49.
- [144] MURPHY, J. J. *Technical analysis of the financial markets: A comprehensive guide to trading methods and applications*. Penguin, 1999.
- [145] N, J. T. Genetic Algorithms Based Combined Strategy Optimization of Select Technical Trading Rules. © 2014 IJIRT | Volume 1 Issue 11 | ISSN: 2349-6002 1, 11 (2014), 137–145.
- [146] NAVET, N., AND CHEN, S. H. On predictability and profitability: Would GP induced trading rules be sensitive to the observed entropy of time series? *Studies in Computational Intelligence* 100 (2008), 197–210.

- [147] NEELY, C. J. Risk-adjusted, ex ante, optimal technical trading rules in equity markets. *International Review of Economics & Finance* 12 (2003), 69–87.
- [148] NETBEANS. NetBeans IDE - Java. <https://netbeans.org/features/java/index.html>. Accedido online en 2019-01-07.
- [149] NOHEJL, A. Grammar-based genetic programming. *Faculty, Prague and Education, Computer Science* (2011).
- [150] NORDIN, P., BANZHAF, W., ET AL. Complexity compression and evolution. In *ICGA* (1995), pp. 310–317.
- [151] NORDIN, P., FRANCONI, F., AND BANZHAF, W. Explicitly defined introns and destructive crossover in genetic programming. *Advances in genetic programming* 2 (1995), 111–134.
- [152] NUTI, G., MIRGHAEMI, M., TRELEAVEN, P., AND YINGSAEREE, C. Algorithmic trading. *Computer* 44, 11 (Nov 2011), 61–69. ISSN 0018-9162.
- [153] O'NEILL, M., AND RYAN, C. Grammatical evolution. *IEEE Transactions on Evolutionary Computation* 5, 4 (2001), 349–358.
- [154] O'NEILL, M., RYAN, C., KEIJZER, M., AND CATTOLICO, M. Crossover in Grammatical Evolution: The Search Continues. In *Genetic Programming: 4th European Conference, EuroGP 2001 Lake Como, Italy, April 18–20, 2001 Proceedings*, vol. 2038. Springer Berlin Heidelberg, 2001, pp. 337–347.
- [155] OPITZ, D., AND MACLIN, R. Popular ensemble methods: An empirical study. *Journal of artificial intelligence research* 11 (1999), 169–198.
- [156] OPITZ, D. W., AND SHAVLIK, J. W. Actively searching for an effective neural network ensemble. *Connection Science* 8, 3-4 (1996), 337–354.
- [157] O'REILLY, U.-M., AND OPPACHER, F. The troubling aspects of a building block hypothesis for genetic programming. In *Foundations of genetic algorithms*, vol. 3. Elsevier, 1995, pp. 73–88.
- [158] OVERLEAF. Overleaf, online latex editor. Disponible en <https://www.overleaf.com/>. Accedido online en 2019-01-07.

- [159] OYEBODE, O. K., AND ADEYEMO, J. A. Genetic programming: principles, applications and opportunities for hydrological modelling. *World Acad Sci Eng Technol Int J Environ Chem Ecol Geol Geophys Eng* 8, 6 (2014), 348–354.
- [160] PEARSON, E., WISHART, J., ET AL. “student’s” collected papers. “*Student’s*” collected papers. (1942).
- [161] PÉREZ LÓPEZ, C., AND CAPELLA, I. *Estadística aplicada a través de Excel*. No. QA278. 8 P4152 2002 in Pearson Education. Prentice Hall, 2002.
- [162] PERKIS, T. Stack-based genetic programming. In *Proceedings of the 1994 IEEE World Congress on Computational Intelligence* (Orlando, Florida, USA, 27-29 June 1994), vol. 1, IEEE Press, pp. 148–153. doi: <https://doi.org/10.1109/ICEC.1994.350025>.
- [163] PERRONE, M. P., AND COOPER, L. N. When networks disagree: Ensemble methods for hybrid neural networks. Tech. rep., BROWN UNIV PROVIDENCE RI INST FOR BRAIN AND NEURAL SYSTEMS, 1992.
- [164] PERSON, J. L. *A complete guide to technical trading tactics : how to profit using pivot points, candlesticks & other indicators*. John Wiley & Sons, 2004.
- [165] PIMENTA, A., NAMETALA, C. A. L., GUIMARÃES, F. G., AND CARRANO, E. G. An automated investing method for stock market based on multiobjective genetic programming. *Computational Economics* (2017), 1–20. ISSN 15729974.
- [166] POLI, R., LANGDON, W., AND MCPHEE, N. *A Field Guide to Genetic Programming*. lulu.com, 2008. ISBN 978-1-4092-0073-4.
- [167] POLI, R., AND LANGDON, W. B. A new schema theorem for genetic programming with one-point crossover and point mutation. *COGNITIVE SCIENCE RESEARCH PAPERS-UNIVERSITY OF BIRMINGHAM CSRP* (1997).
- [168] POLI, R., MCPHEE, N. F., AND ROWE, J. E. Exact schema theory and markov chain models for genetic programming and variable-length genetic algorithms with homologous crossover. *Genetic Programming and Evolvable Machines* 5, 1 (2004), 31–70.
- [169] POLIKAR, R. Ensemble based systems in decision making. *IEEE Circuits and systems magazine* 6, 3 (2006), 21–45.

- [170] POTVIN, J. Y., SORIANO, P., AND MAXIME, V. Generating trading rules on the stock markets with genetic programming. *Computers and Operations Research* 31, 7 (2004), 1033–1047.
- [171] R. R: What is R? <https://www.r-project.org/about.html>. Accedido online en 2019-01-07.
- [172] RAS, G., HASELAGER, P., AND GERVEN, M. V. Explanation methods in Deep Learning: What and how much do they explain? *arXiv:1803.07517* (2018), 1–15.
- [173] RAUL CANESSA C. Los Puntos Pivote | Análisis Técnico | Técnicas de Trading. <https://www.tecnicasdetrading.com/2009/07/puntos-pivote.html> <http://www.tecnicasdetrading.com/2009/07/puntos-pivote.html>, 2018. On line en 2019-02-19.
- [174] RAUL CANESSA C. Puntos pivote de Fibonacci | Técnicas de Trading. <https://www.tecnicasdetrading.com/2014/06/puntos-pivote-de-fibonacci.html>, 2018. On line en 2019-02-19.
- [175] RECHENBERG, I. *Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*. Fromman-Holzbook, Stuttgart, Germany, 1973.
- [176] RECHENBERG, I. *Evolutionsstrategie'94*, volume 1 of werkstatt bionik und evolutionstechnik. *Frommann Holzboog, Stuttgart* (1994).
- [177] ROKACH, L. Ensemble-based classifiers. *Artificial Intelligence Review* 33, 1-2 (2010), 1–39.
- [178] ROLLINGER, T., AND HOFFMAN, S. Sortino ratio: A better measure of risk. *Futures Magazine* 1, 02 (2013).
- [179] RUSSELL, S., AND NORVIG, P. *Artificial intelligence: a modern approach*. 3rd. *Essex, UK: Parentice Hall* (2009), 1152.
- [180] RYAN, C. Pygmies and civil servants. In *Advances in Genetic Programming* (1994), MIT Press, pp. 243–263.
- [181] RYAN, C., COLLINS, J., AND O'NEIL, M. Grammatical Evolution: Evolving Programs for an Arbitrary Language. *Springer Berlin Heidelberg* (1998), 83–96.

- [182] SALUSTOWICZ, R., AND SCHMIDHUBER, J. Probabilistic incremental program evolution. *Evolutionary Computation* 5, 2 (jun 1997), 123–141. doi: <https://doi.org/10.1162/evco.1997.5.2.123> ISSN 10636560 ISBN 3540628584.
- [183] SCHMIDBAUER, H., RÖSCH, A., SEZER, T., AND TUNALIOĞLU, V. S. Robust trading rule selection and forecasting accuracy. *Journal of Systems Science and Complexity* 27, 1 (2014), 169–180.
- [184] SCHMIDHUBER, J. Deep learning in neural networks: An overview. *Neural networks* 61 (2015), 85–117.
- [185] SCHWEFEL, H.-P. Nichtlineare optimierung von computermodellen mittels evolutionsstrategie, 1977.
- [186] SCHWEFEL, H.-P. *Numerical Optimization of Computer Models*. John Wiley & Sons, Inc., New York, NY, USA, 1981. ISBN 0471099880.
- [187] SETZKORN, C., DIPIETRO, L., AND PURSHOUSE, R. Evolving Rule-Based Trading Systems. *36th Annual Meeting of the CEA* (2002).
- [188] SHAPIRO, S. S., AND WILK, M. B. An analysis of variance test for normality (complete samples). *Biometrika* 52, 3/4 (1965), 591–611.
- [189] SHARELATEX. ShareLaTeX, Editor de LaTeX online. Disponible en <https://es.sharelatex.com/>. Accedido online en 2019-01-07.
- [190] SHARPE, W. F. Mutual fund performance. *The Journal of business* 39, 1 (1966), 119–138.
- [191] SHEHAB, M., KHADER, A. T., AND AL-BETAR, M. New selection schemes for particle swarm optimization. *IEEJ Transactions on Electronics, Information and Systems* 136, 12 (2016), 1706–1711.
- [192] SHI, Y., AND EBERHART, R. A modified particle swarm optimizer. In *Evolutionary Computation Proceedings, 1998. IEEE World Congress on Computational Intelligence., The 1998 IEEE International Conference on* (1998), IEEE, pp. 69–73.
- [193] SHUKLA, A., PANDEY, H. M., AND MEHROTRA, D. Comparative review of selection techniques in genetic algorithm. *2015 1st International Conference on Futuristic Trends in Computational Analysis and Knowledge Management, ABLAZE 2015*, February (2015), 515–519.

- [194] SIEGAL, S. *Nonparametric statistics for the behavioral sciences*. McGraw-hill, 1956.
- [195] SIERRA BRAVO, R. *Técnicas de Investigación Social: Teoría y Ejercicios*. Paraninfo, 2001. ISBN 8428324298.
- [196] SIVANANDAM, S., AND DEEPA, S. Genetic Algorithm Implementation Using Matlab. *Introduction to Genetic Algorithms* (2008), 211–262.
- [197] SIVANANDAM, S. N., AND DEEPA, S. N. *Introduction to genetic algorithms*. Springer Berlin / Heidelberg, Berlin, Heidelberg, 2008.
- [198] SIVARAJ, R., AND RAVICHANDRAN, T. A review of selection methods in genetic algorithm. *International Journal of Engineering Science and Technology (IJEST)* 3, 5 (2011), 3792–3797. doi: <https://doi.org/10.18535/ijecs/v6i12.04> ISSN 0975-5462, ISBN 0975-5462.
- [199] SOULE, T., AND FOSTER, J. A. Removal bias: a new cause of code growth in tree based evolutionary programming. In *1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No.98TH8360)* (1998), pp. 781—786. doi: <https://doi.org/10.1109/ICEC.1998.700151> ISBN 0-7803-4869-9.
- [200] SPECTOR, L. Simultaneous evolution of programs and their control structures. *Advances in genetic programming* 2 (1996), 137–154.
- [201] SPIEGEL, M. R., AND STEPHENS, L. J. *Estadística/Theory and problems of statistics*. No. 519.2 in Schaum. McGraw-Hill,, 2009.
- [202] STORN, R., AND PRICE, K. Differential evolution—a simple and efficient adaptive scheme for global optimization over continuous spaces [r]. *Berkeley: ICSI* (1995).
- [203] SUGENO, M. *Industrial applications of fuzzy control*. Elsevier Science Inc., 1985.
- [204] SWEENEY, R. J., AND LEE, E. J. International dimensions of securities and currency markets. *Advances in financial planning and forecasting series 4, Part A* (1990).
- [205] SYSWERDA, G. Uniform crossover in genetic algorithms. In *Proceedings of the third international conference on Genetic algorithms* (1989), Morgan Kaufmann Publishers, pp. 2–9.

- [206] THOMAS, J. D., AND SYCARA, K. {GP} and the Predictive Power of Internet Message Traffic. In *Genetic Algorithms and Genetic Programming in Computational Finance*. Springer US, Boston, MA, 2002, pp. 81–102.
- [207] THORHAUER, A., AND ROTHLAUF, F. Structural difficulty in grammatical evolution versus genetic programming. In *Proceedings of the 15th annual conference on Genetic and evolutionary computation - GECCO '13 (2013)*, p. 997.
- [208] TIAN, X. Optimization of Intraday Trading Strategy Based on ACD Rules and Pivot Point System in Chinese Market. *Journal of Intelligent Learning Systems and Applications* 04, 04 (2012), 279–284. doi: <http://dx.doi.org/10.4236/jilsa.2012.44029> ISSN 2150-8402, ISBN 978-1-4503-0557-0.
- [209] TOPCHY, A., AND PUNCH, W. F. Faster genetic programming based on local gradient search of numeric leaf values. In *Proceedings of the 3rd Annual Conference on Genetic and Evolutionary Computation (2001)*, Morgan Kaufmann Publishers Inc., pp. 155–162.
- [210] UNIVERSITY GEORGE MASON. A Java-based Evolutionary Computation Research System, ECLab Evolutionary Computation Laboratory. <https://cs.gmu.edu/~eclab/projects/ecj/> email: [cj-helpcs.gmu.edu](mailto:cj-helpcs.gmu.edu), 2018 (accedido 19 diciembre 2018).
- [211] WELCH, B. L. The Generalization of Student's Problem When Several Different Population Variances Are Involved. *Biometrika* 34, 1-2 (jan 1947), 28–35. doi: <https://doi.org/10.1093/biomet/34.1-2.28>, ISSN 00063444.
- [212] WHITLEY, D. Genitor: A different genetic algorithm. In *Proceedings of the Rocky Mountain conference on artificial intelligence, 1988 (1988)*, pp. 118–130.
- [213] WHITLEY, D., GORDON, V. S., AND MATHIAS, K. Lamarckian evolution, the baldwin effect and function optimization. In *International Conference on Parallel Problem Solving from Nature (1994)*, Springer, pp. 5–15.
- [214] WILCOXON, F., PAIARDINI, A., PISCITELLI, M., PASCARELLA, S., BECHORSHENTAL, D., MARTZ, E., BEN-TAL, N., BOURNE, P., ESCLAPEZ, J., AND RICE, D. Individual Comparisons by Ranking Methods. *Biometrics Bulletin* 1, 6 (1945), 80. doi: <https://doi.org/10.2307/3001968>, ISBN 0099-4987, ISSN 00994987.



- [215] WILDER, J. W. *New concepts in technical trading systems*. Trend Research, 1978.
- [216] WILIŃSKI, A., NYCZAJ, T., BERA, A., AND BŁASZYŃSKI, P. A study on the effectiveness of investment strategy based on the concept of pivot points levels using Matthews criterion. *Journal of Theoretical and Applied Computer Science* (2013).
- [217] WILSON, G., LEBLANC, D., AND BANZHAF, W. Stock trading using linear genetic programming with multiple time frames. *GECCO '11: Proceedings of the 13th annual conference on Genetic and evolutionary computation* (2011), 1667–1674. doi: <https://doi.org/10.1145/2001576.2001801>, ISBN 9781450305570.
- [218] WOLPERT, D. H., MACREADY, W. G., ET AL. No free lunch theorems for search. Tech. rep., Technical Report SFI-TR-95-02-010, Santa Fe Institute, 1995.
- [219] WRIGHT, S. *The roles of mutation, inbreeding, crossbreeding, and selection in evolution*, vol. 1. na, 1932.
- [220] WRIGHT, S. Surfaces of selective value revisited. *The American Naturalist* 131, 1 (1988), 115–123.
- [221] YANG, S. Y., MO, S. Y. K., LIU, A., AND KIRILENKO, A. A. Genetic programming optimization for a sentiment feedback strength based trading strategy. *Neurocomputing* 264 (2017), 29–41.
- [222] ZHOU, Z.-H. *Ensemble methods: foundations and algorithms*. Chapman and Hall/CRC, 2012.

*Esta página se ha dejado en blanco intencionadamente.*