IJCSMS International Journal of Computer Science & Management Studies, Special Issue of Vol. 12, June 2012
ISSN (Online): 2231 –5268
www.ijcsms.com

Keel A Data Mining Tool: Analysis With Genetic

Manju Narwal[1], Ms. Pooja Mittal[2]

[1]M.tech Student, Department of Computer Science and Applications,
M. D. University, Rohtak-124001, Haryana, India
mnarwal87@gmail.com

[2]Assistant Professor, Department of Computer Science and Applications,
M. D. University, Rohtak-124001, Haryana, India

Abstract
This work is related to the KEEL (Knowledge Extraction based on Evolutionary Learning) tool, an open source software that supports data management and provides a platform for the analysis of evolutionary learning for Data Mining problems of different kinds including as regression, classification, unsupervised learning. It includes a big collection of evolutionary learning algorithms based on different approaches: Pittsburgh, Michigan. It empowers the user to perform complete analysis of any genetic fuzzy system in comparison to existing ones, with a statistical test module for comparison.

Keywords: *Genetic programming, Data mining, Evolutionary algorithms, Experimental design, Graphical programming, Java Knowledge extraction, Machine learning.*

Introduction

Evolutionary Algorithms (EAs) are optimization algorithms based on natural evolution and genetic processes. In Artificial Intelligence (AI), EAs are one of the most successful search techniques for complex problems. Recently EAs, particularly Genetic Algorithms (GAs) have proved to be an important technique for learning and knowledge extraction. This makes them also a promising tool in Data Mining .The idea of automatically discovering knowledge from databases is a very attractive and complex task. That's why, there has been a growing interest in DM in various AI related areas, including EAs. The main objective for applying EAs to knowledge extraction tasks is that they are robust and adaptive search methods that perform a global search in place of candidate solutions. The use of EAs in problem solving such as image retrieval, the learning of controllers in robotics and the improvement of E-learning systems show their suitability.

In a wide range of scientific fields as a problem solver EAs are powerful for solving a wide range of scientific problems, their use requires a certain programming expertise along with considerable time and effort to write a computer program for implementing the often sophisticated algorithm according to user needs. This work can be tedious and needs to be done before users can start focusing their attention on the issues that they should be really working. For this given situation, the aim of this paper is to introduce a non-commercial Java software tool named *KEEL* (Knowledge Extraction based on Evolutionary Learning). This tool empowers the user to analyze the behavior of evolutionary learning for various kinds of DM problems: regression, classification, unsupervised learning etc.

This tool can provide several benefits. First of all, it reduces programming work. It includes a library with evolutionary learning algorithms based on different paradigms (Pittsburgh, Michigan and IRL) and simplifies the integration of evolutionary learning algorithms with different preprocessing techniques. It can alleviate researchers from the mere "technical work" of programming and enable them to focus more on the analysis of their new learning models in comparison with the existing ones. Secondly, it extends the range of possible users applying evolutionary learning algorithms. An extensive library of EAs together with easy- to-use software considerably reduces the level of knowledge and experience required by researchers in evolutionary computation. As a result researchers with less knowledge, when using this framework, would be able to apply successfully these algorithms to their problems. Third, due to the use of a strict object-oriented approach for the library and software tool, these can be used on any machine with Java. As a result, any researcher can use KEEL on his machine, independently of the operating system.

This paper is arranged as follows. The next section introduces a study on Genetic fuzzy systems Section 3)

IJCSMS International Journal of Computer Science & Management Studies, Special Issue of Vol. 12, June 2012
ISSN (Online): 2231 –5268
www.ijcsms.com

presents KEEL: its main features and modules. In Section 4) data management describe with example how data management is done within KEEL. Finally, Section 5 points out some conclusions and future work.

Genetic Fuzzy Systems

Computational Intelligence techniques such as artificial neural networks, fuzzy logic, and genetic algorithms (GAs) are popular research subjects, since they can deal with complex engineering problems which are difficult to solve by classical methods. So Hybrid approaches have attracted considerable attention in the Computational Intelligence community.

One of the most popular approaches is the hybridization between Fuzzy Logic and GAs leading to genetic fuzzy systems (GFSs). GFS is basically a fuzzy system augmented by a learning process based on a GA.GAs are search algorithms based on natural genetics that provide robust search capabilities in complex spaces, and there by offer a valid approach to problems requiring efficient and effective search processes. Fuzzy systems are one of the most important areas for the application of the Fuzzy Set Theory. Fuzzy systems have been successfully applied to solve different kinds of problems in various application domains.
In this contribution we introduce a non-commercial Java software tool named KEEL (Knowledge Extraction based on Evolutionary Learning). This tool empowers the user to assess the behavior of EAs for different kinds of Data Mining problems: regression, classification, clustering, pattern mining, etc. Consequently, the application of EAs for learning fuzzy systems is also included in KEEL, including a representative set of GFSs. It allows us to perform a complete analysis of any genetic fuzzy system in comparison to existing ones, including a statistical test module for comparison. This tool can offer several advantages:
First of all, it reduces programming work. It includes a big library with GFS algorithms based on different paradigms (Pittsburgh, Michigan, IRL and GCCL) and simplifies their integration with different pre-processing techniques. It can alleviate researchers from the mere "technical work" of programming and enable them to focus more on the analysis of their new learning models in comparison with the existing ones.
Secondly, it extends the range of possible users applying GFSs. A library of algorithms together with easy-to-use software considerably reduces the level of knowledge and experience required by researchers in evolutionary computation and fuzzy logic. As a result researchers with less knowledge, when using this framework, would be able to apply successfully these algorithms to their problems.
Third, due to the use of a strict object-oriented approach for the library and software tool, these can be used on any machine with Java. As a result, any researcher can use KEEL on his machine, independently of the operating system.

Keel Description

KEEL is a software tool to assess EAs for DM problems including regression, classification, clustering and pattern mining and so on. The presently available version of KEEL consists of the following function blocks**.**

1) Data Management -- This part is composed of a set of tools that can be used to build new data, export and import data in other formats to the KEEL format, data edition and visualization, applying transformations and partitioning to data, etc.

2) Design of Experiments -- The aim of this part is the design of the desired experimentation over the selected data sets. It provides many options to choose from: type of validation, type of learning (classification, regression, unsupervised learning), etc.

3) Educational Experiments -- With a similar structure to the previous part, it allows you to design an experiment which can be debugged step-by-step in order to use this as a guideline, to show the learning process of a certain model by using the platform for educational objectives.

Taking into account each one of the above function blocks, KEEL can be useful for different types of users, who expect to find determined features in Data Mining (DM) software.

The following describes the ‘User Profiles’ of who KEEL is designed for, its Main Features and the different ways of working with it.

Keel User Profiles

KEEL is an integration of an environment with a defined architecture and the development of knowledge extraction as expandable modules. It is mainly intended for two (2) categories of users: researchers and students. Either group has a different set of needs:

1) KEEL as a research tool -- The most common use of this tool for a researcher will be the automated execution of experiments, and the statistical analysis of their results. Routinely, an experimental design includes a mix of evolutionary algorithms, statistical and Artificial Intelligent related techniques. Special care has been taken to make it possible for a researcher to use KEEL to assess the relevance of their own procedures.

IJCSMS International Journal of Computer Science & Management Studies, Special Issue of Vol. 12, June 2012
ISSN (Online): 2231 –5268
www.ijcsms.com

Since the actual standards in machine learning require heavy computational work, the research tool is not designed to offer a real-time view of the progress of the algorithms, it is designed to generate a script and be batch-executed in a cluster of computers.

The tool allows the researcher to apply the same sequence of pre-processing, experiments and analysis to large batteries of problems and focus their attention to the summary of the results.

2) KEEL as an educational tool -- The needs of a student are quite different to those of a researcher. Generally speaking, the objective is No longer that of making statistically sound comparisons between algorithms. There is No need of repeating each experiment a large number of times. If the tool is to be used in a class, the execution time must be short and a real-time view of the evolution of the algorithms is needed, since the student will use this information to learn how to adjust the parameters of the algorithms. In this sense, the educational tool is a simplified version of the research tool, where only the most relevant algorithms are available. The execution is made in real time. The user has visual feedback of the progress of the algorithms and can access the final results from the same interface used to design the experiment.

Main Features of Keel --

KEEL is a software tool developed to assemble and use different Data Mining models. KEEL is one of the first software toolkits of its type that contains a library of evolutionary learning algorithms with open source code in Java. The main features of KEEL are:

1) Evolutionary Algorithms (EAs) are presented in predictive models, pre-processing (evolutionary feature and instance selection) and post-processing (evolutionary tuning of fuzzy rules).

2) It includes data pre-processing algorithms proposed in specialized literature: data transformation, discretization, instance selection and feature selection.

3) It has a statistical library to analyze an algorithms result. It consists of a set of statistical tests for analyzing the normality and heteroscedasticity of the results and performs parametric and non-parametric comparisons among the algorithms.

4) Some of the algorithms have been developed with the Java Class Library for Evolutionary Computation (JCLEC).

5) KEEL provides a user-friendly interface, oriented to the analysis of algorithms.

6) The software's aim is to create experiments containing multiple data sets and algorithms connected together to obtain an expected result. Experiments are independently script-generated from the 'user interface' for an off-line run in the same or in other machines.

7) KEEL also allows you to create experiments in on-line mode, aimed at educational support, in order to learn the operation of the algorithms included.

8) KEEL contains a 'Knowledge Extraction Algorithms Library', consisting of the incorporation of multiple evolutionary learning algorithms with classical learning approaches. The main library features include:

- a) Different evolutionary rule learning models have been implemented.
- b) Fuzzy rule learning models with a good trade-off between accuracy and interpretability.
- c) Evolution and pruning in neural networks, product unit neural networks, and radial base function (RBF) models.
- d) Genetic Programming: Evolutionary algorithms that use tree representations for extracting knowledge.
- e) Algorithms for extracting descriptive rules based on pattern(s) subgroup discovery have been integrated.
- f) Data reduction (instance and feature selection and discretization). EAs for data reduction have also been included.

9) Keel software operates via a web interface, allowing end-user access from all web-enabled computers.

Three recent new aspects/features of KEEL --

1) KEEL-dataset, a data set repository that includes the data set partitions in the KEEL format and also shows some results of the algorithms in these data sets. This repository can free researchers from merely doing "technical work" and makes the comparison of their models with existing models easier.

2) KEEL has been developed with the idea of being easily extended with new algorithms. For this reason, the manufacturer introduces some basic guidelines that the developer may take into account for managing the specific constraints of the KEEL tool. Moreover, a source code template has been made available to manage all the restrictions of the KEEL software, including the input and

IJCSMS International Journal of Computer Science & Management Studies, Special Issue of Vol. 12, June 2012
ISSN (Online): 2231 –5268
www.ijcsms.com

output functions, the parsing of the parameters, and the class structure. The manufacturer describes in detail this template showing a simple algorithm, the "Steady-State Genetic Algorithm for Extracting Fuzzy Classification Rules from Data" (SGERD) procedure (see published paper below...).

3) A module of statistical procedures was developed in order to provide the researcher with a suitable tool to contrast the results obtained in any experimental study performed inside the KEEL environment.

The manufacturer describes this module and shows a case study using some non-parametric statistical tests for the multiple comparison of the performance of several 'genetic rule' learning methods for classification.

Data management

The fundamental purpose of data preparation is to manipulate and transform raw data so that the information content enfolded in the data set can be exposed, or made more accessible. Data preparation comprises those techniques concerned with analyzing raw data so as to yield quality data, mainly including data collecting, data integration, data transformation, data cleaning, data reduction and data discretization .Data preparation can be even more time consuming than data mining, and can present equal challenges to data mining. Its importance lies in that the real-world data is impure (incomplete, noisy and inconsistent) and high-performance mining systems require quality data (the removal of anomalies or duplications). Quality data yields high-quality patterns (to recover missing data, purify data and resolve conflicts).

The Data Management module integrated in KEEL allows us to perform the data preparation stage independently of the remaining of the DM process itself. This module is focused on the group of users denoted as domain experts. They are familiar with their data, they know the processes that produce the data and they are interested in reviewing those to improve upon or analyze them. On the other hand, domain users are those whose interest lies in applying processes to their own data and they usually are not experts in DM.

The next figure shows an example window of the *Data Management* module in the section of *Data Visualization.* The module has seven sections, each of which is accessible through the buttons on the left side of the window. In the following, we will briefly describe them:

- *Creation of a new data set*: This option allows us to generate a new data set compatible with the other KEEL modules.
- *Import data to KEEL format*: Since KEEL works with a specific data format (alike the ARFF format) in all its modules, this section allows us to convert various data formats to KEEL format, such as CSV, XML, ARFF, extracting data from data bases, etc.
- *Export data from KEEL format*: This is the opposite option to the previous one. It converts the data handled by KEEL procedures in other external formats to establish compatibility with other software tools.
- *Visualization of data*: This option is used to represent and visualize the data. With it, we can see a graphical distribution of each attribute and comparisons between two attributes.
- *Edition of data*: This area is dedicated to managing the data manually. The data set, once loaded, can be edited by terms of modifying values, adding or removing rows and columns, etc.
- *Data Partition*: This zone allows us to make the partitions of data needed by the experiment modules to validate results. It supports *k*-fold cross validation, 5×2 cross validation and hold-out validation with stratified partition.

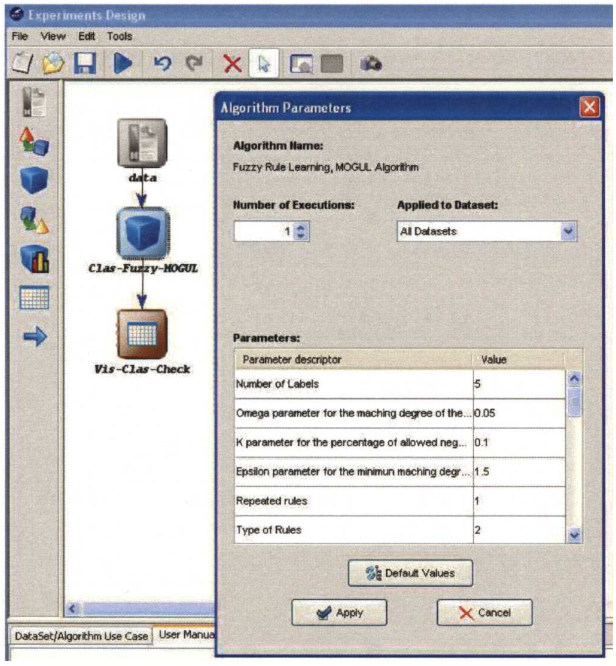
- *Data Preparation*: This section allows us to perform automatic data preparation for DM, including cleaning, transformation and reduction of data. All techniques integrated in this section are also available in the experiments-related modules.

The experiments can be graphically modeled, on the basis of data flow and represented by graphs with node-edge connections. It allows us to choose the type of validation (k-fold cross validation or 5x2 cross validation) and type of learning (regression, classification or unsupervised).
Then, we have to select the data sources, drag the selected methods into the workspace and establish connections between methods and data sets. Also, the addition of statistical analysis of results is supported by including the corresponding techniques. In any moment, each component added in the experiment can be configured by double-clicking in the respective node. The below mentioned figure shows an example of an experiment following the MOGUL methodology in classification and using a report box to obtain a summary of results.

When the experiment has been designed, the user can choose either to save the design in XML file or obtain a zip file. The latter will contain the directory structure and required files for running the experiment in an independent machine with Java. The zip file will include the data sources, jar files of the algorithms, configuration files in XML format and a Java Tool, named Run Keel, to run the experiment. Run Keel can be seen as a simple scripting environment that interprets the script file in XML format,

IJCSMS International Journal of Computer Science & Management Studies, Special Issue of Vol. 12, June 2012
ISSN (Online): 2231 –5268
www.ijcsms.com

runs all the indicated algorithms and saves the results in one or several report files.



RESULTS:

System Implementation is used to bring a developed system or sub system into operational use and turning it over to the user. It involves programmer users and operational managements.
System Implementation components include:
Personal Orientation: Introduce people to the new system and their relationship to the system.
Training: Give employees the tools and techniques to operate and use the system.
Hardware Installation: Schedule for, prepare for, and then actually install new equipment.
Procedure Writing: Develop procedure manual to follow in operating the new system.
Testing: Ensure that the computer programs properly process the data.
File Conversion: Load the information of the present files into the new system files.
Parallel Operation: Use the new system at the same time as the old to make sure results are.

5.1 SCREENS

The user has to prepare one Dataset, then he has to import the dataset means converting that dataset in to keel format.Now he can use that converted dataset to perform the operations of KEEL on it.After running the experiment it will display as Experiment is Successfully generated ,then wecan see the result of the experiment in the destination folder.Now we will see the screen shots of above discussion to understand better.

Fig 5.1 Home page of KEEL

- The Data Management module maintains datasets
- The Experiments module performs experiment on datasets
- Educational experiments show the statistical results
- Help gives the basic information of KEEL

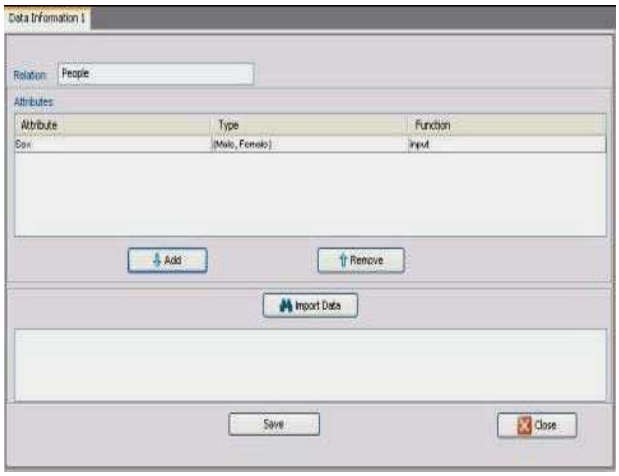


Fig 5.2 preparing a Dataset

IJCSMS International Journal of Computer Science & Management Studies, Special Issue of Vol. 12, June 2012
ISSN (Online): 2231 –5268
www.ijcsms.com

- After clicking the new Dataset the above screen will appear

Fig 5.3 Adding attributes

- Press the add button to add attributes

Fig 5.4 Importing a Dataset

- Select a Dataset which you want to convert
- Select a destination folder where you want to store
- Click the convert button to convert in to KEEL format

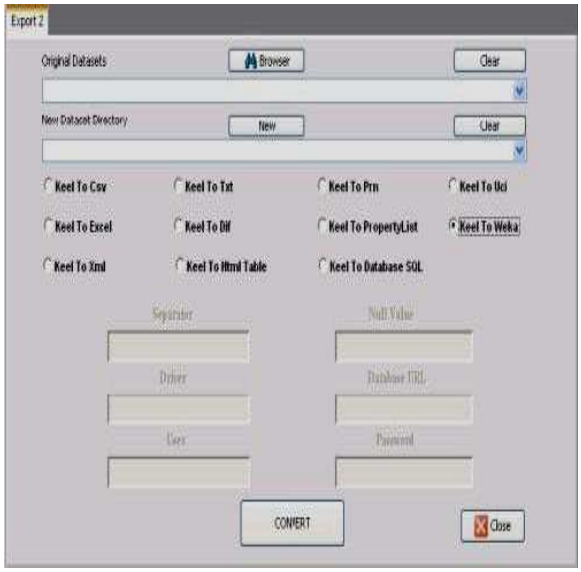


Fig 5.5 Exporting a Dataset

- Select a KEEL Dataset you want to convert
- Select a destination folder you want to store Dataset
- Click on convert to convert in to normal form

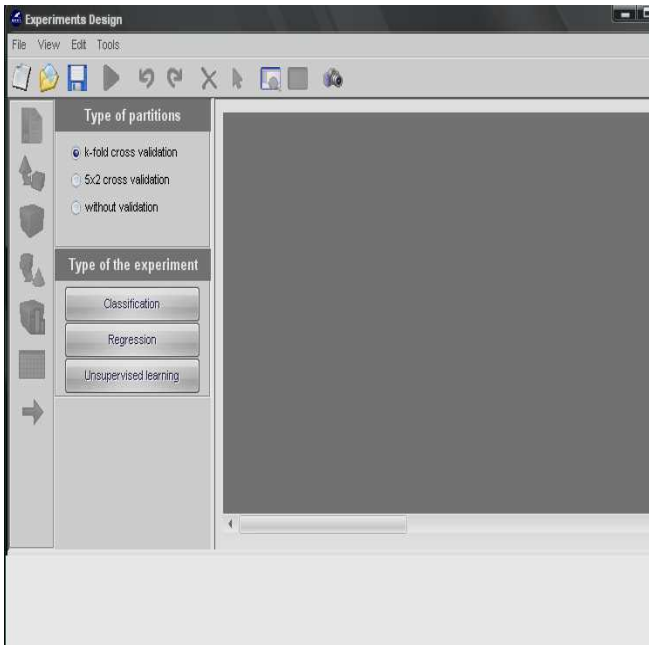


Fig 5.6 Selection of Experiment

- Select the type of validation
- Select the type of experiment

IJCSMS International Journal of Computer Science & Management Studies, Special Issue of Vol. 12, June 2012
ISSN (Online): 2231 –5268
www.ijcsms.com

Fig 5.7 selecting the Dataset for Experiment

- Select the Dataset for experiment
- Click on the Dataset and drop on the experiment area

Fig 5.8 Running the Experiment

- Select the type of algorithms
- Connect it to Dataset
- Then run the Experiment

Fig 5.9 Result Window

- After completing the Experiment it will show the window as above
- The resultant folder contains Exe, Scripts, Datasets, Results

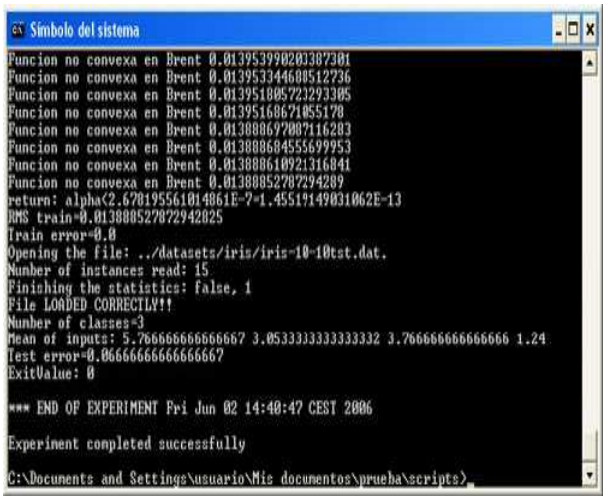


Fig 5.10 Result screen in command prompt

- After completing the experiment it will show as Experiment completed successfully in command prompt

IJCSMS International Journal of Computer Science & Management Studies, Special Issue of Vol. 12, June 2012
ISSN (Online): 2231 –5268
www.ijcsms.com

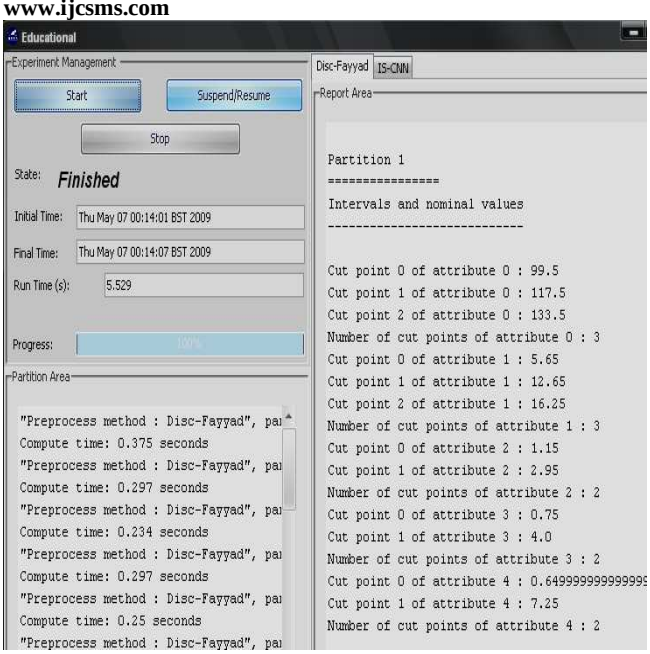


Fig 5.11 Statistical Analysis of Result

- After educational experiment it will show the statistical result, like initial time final time number of partitions etc..

CONCLUSIONS

In this work, we have described KEEL, a software tool to Assess EAs for DM problems, paying special attention to the GFS algorithms integrated in the tool. It relieves researchers of much technical work and allows them to focus on the analysis of their new GFS algorithms in comparison with the existing ones. Moreover, the tool enables researchers with a basic knowledge of fuzzy logic and evolutionary computation to apply GFSs to their work. We have shown Data management to illustrate functionalities and the experiment set up processes in KEEL. The KEEL software tool is being continuously updated and improved. At the moment, we are developing a new set of GFSs and a test tool that will allow us to apply parametric and non-parametric tests on any set of data. We are also developing data visualization tools for the on-line and offline modules.

REFERENCES

[1] D.E. Goldberg, Genetic algorithms in search, optimization, and machine Learning, Addison-Wesley Professional, Canada (1989)
[2] J.H. Holland, Adaptation in natural and artificial systems, Ann Arbor: University of Michigan Press (1975).
[3] O. Cord´on, F. Herrera, F. Hoffmann and L. Magdalena, Genetic fuzzy systems Evolutionary tuning and learning of fuzzy knowledge bases, World Scientific, Singapore (2001)
[4] A.E. Eiben and J.E. Smith, Introduction to Evolutionary Computing.
[5] S. Smith, A learning system based on genetic algorithms, Ph.D. thesis, University of Pittsburgh (1980).
[6] K.A. De Jong, W.M. Spears and D.F. Gordon, Using genetic algorithms for concept learning, Machine Learning 13 (1993) 161-188
[7] S.W. Wilson, Classifier Fitness Based on Accuracy, Evolutionary Computation 3:2 (1995) 149-175.
[8] E. Bernad´o-Mansilla and J.M. Garrell, Accuracy-Based Learning Classifier Systems: Models, Analysis and Applications to Classification Tasks, Evolutionary Computation 11:3 (2003) 209-238.
[9] O. Cord´on and F. Herrera, Hybridizing Genetic Algorithms with Sharing Scheme and Evolution Strategies for Designing Approximate Fuzzy Rule-Based Systems, Fuzzy Sets and Systems 118:2 (2001) 235- 255.
[10] L.X. Wang and J.M. Mendel, Generating Fuzzy Rules by Learning from Examples, IEEE Transactions on Systems, Man and Cybernetics 22:6 (1992) 1414-1427.