On Semi-Supervised Clustering via Multiobjective Optimization

Julia Handl
Manchester Interdisciplinary Biocentre
University of Manchester, UK
j.handl@postgrad.manchester.ac.uk

Joshua Knowles
Manchester Interdisciplinary Biocentre
University of Manchester, UK
j.knowles@manchester.ac.uk

ABSTRACT

Semi-supervised classification uses aspects of both unsupervised and supervised learning to improve upon the performance of traditional classification methods. Semi-supervised *clustering*, in particular, explicitly integrates both information about the data distribution and about class memberships into the clustering process. In this paper, the potential of a multiobjective formulation of the semi-supervised clustering problem is explored, and two evolutionary multiobjective approaches to the problem are outlined. Experimental results demonstrate practical performance benefits of this methodology, including an improved classification performance and an increased robustness towards annotation errors.

Categories and Subject Descriptors: I.5 [Pattern Recognition]: Clustering

General Terms: Algorithms.

Keywords: semi-supervised clustering, multiobjective clustering, multiobjective machine learning, semi-supervised learning.

1. INTRODUCTION

Existing machine learning algorithms differ along a number of different dimensions, one of the most fundamental of which is the distinction between unsupervised and supervised learning techniques. Supervised learning refers to learning in the presence of training examples—in classification, a set of data samples for which the correct classification is known. If a sufficient amount of such data is available, a classifier can be trained in order to learn and correctly predict the class memberships of these data items in the hope that the trained classifier subsequently generalizes to the classification of new unlabeled data. Supervised methods can be very powerful for the classification of complex data, but may suffer from problems related to overtraining, resulting in poor generalization capabilities.

In contrast to supervised learning, unsupervised learning can be applied in the absence of any prior knowledge about the number of classes, or any correct training examples. It relies on the assumption that the main class structure of the data is reflected by the actual distribution of the data, that is, that clusters of homogeneous data items can be identified and that this grouping will lead to a meaningful classification. Evidently, unsupervised approaches are prone to fail if no distinct cluster structure is present in the data, and they are, in this sense, less powerful than supervised methods. However, their positive aspects include the facts that, in contrast to supervised approaches, they can be used for exploratory data analysis in scenarios where little prior information is given and that they are not affected by overtraining.

All things considered, unsupervised and supervised learning approaches must be seen as complementary approaches to the task of data classification, whose respective success and applicability depends on the features of the problem domain. But a 'third' or 'middle way' also exists — *semi-supervised classification* (described in detail in the next section), which is able to combine the advantages of its two older brothers in some cases. In this paper, we review the current state of semi-supervised classification and motivate a multiobjective approach to semi-supervised clustering. Section 3 goes on to describe two alternative implementations of the approach, and Section 4 describes the main research questions addressed in this work as well as the corresponding experimental setup. The results of our experiments are presented in Section 5 and, finally, Section 6 discusses the implications of our findings and concludes.

2. SEMI-SUPERVISED CLASSIFICATION

In certain classification scenarios it can be advantageous to combine the advantages of both unsupervised and supervised classification techniques, that is, to exploit both previous knowledge of class labels and the underlying data distribution: semi-supervised approaches aim to do this. Through the combined use of labeled and unlabeled data it becomes possible to give a degree of external guidance to the classification algorithm, while still permitting intrinsic structure in the data to be taken into account. This is considered particularly useful when dealing with data sets consisting of a large number of unlabeled data items and relatively few labeled ones, and, more generally, in the case of very limited prior knowledge. For example, in cases where the classes within a particular data set are only partially known, additional ones may be identified by taking the data distribution into account (see Figure 1). Also, due to the

GECCO'06, July 8–12, 2006, Seattle, Washington, USA.

combination of two fundamentally different sources of information, semi-supervised approaches would be expected to be more robust than both unsupervised and supervised approaches, and may be less sensitive towards both annotation errors and the occlusion of structures in the data due to noise.

Data sets with the above properties are frequently encountered in application domains where the categorization of individual data items is accompanied by high computational, analytical or experimental costs. Initially introduced in the field of information retrieval, semi-supervised methods have now seen first application in post-genomic data analysis, in particular for gene function classification [14, 15], protein classification [21] and the prediction of patient survival from gene expression data [1]. In general, semi-supervised classification promises to be a fruitful approach in bioinformatics, as the availability of only few labeled samples married with large amounts of unlabeled data is the general rule in that field.

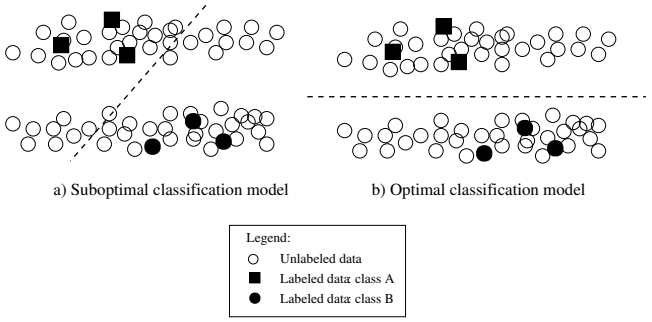

Figure 1: Illustration of the fundamental idea behind semi-supervision. The unlabeled data points can help to avoid suboptimal solutions and identify the classification model that is optimal with respect to the given data.

2.1 Transductive inference

Semi-supervised learning is closely related to the principle of transduction, which has been described as an additional principle of inference by Vapnik [20].

The three most established types of human inference are deduction, induction and abduction. Of these, only deduction is based on the rules of formal logic, and its conclusions are necessarily true. In contrast, both inductive and abductive reasoning, which can be considered as counterparts to deductive reasoning, lead to uncertain predictions: an inductive argument takes a number of observations, consisting of individual cases and the results associated, and attempts to predict a general rule that relates the cases with their results; abductive reasoning employs a rule and an observed result and hypothesises that this particular result is an instance of the application of the rule, and that the antecedent of the rule is therefore true. The ability to deal with uncertainty and to perform generalization make induction and abduction powerful paradigms in the context of machine learning.

Differently to induction, transduction generalizes from observed, specific cases to other specific cases, and *not* to general principles. Thus, methods of transduction do not attempt to develop a general model that can subsequently be used for deduction, but they use inference from case to case. Vapnik argues that this avoids the solution of a more general problem (the inference of an unobserved model) before solving a more specific problem (the deduction of the results for new cases).

2.2 Previous work on semi-supervised learning

The shared concept between all techniques of semi-supervision is the fact that two types of information sources are exploited: (i) partial knowledge about the class memberships, which can be given in the form of class labels or "must-link" and "cannot-link" constraints; as well as (ii) knowledge of the underlying data distribution, which is used under the assumption of coherence in data-space , that is, the assumption that close neighbours in data space are likely to have identical class membership. Importantly, semi-supervised classification is based on the idea that unsupervised and supervised information are consistent and complement each other and that their combined use can therefore improve classification accuracy. Consequently, no improvement can be expected if these assumptions are (strongly) violated.

Two different approaches to semi-supervised learning can be distinguished, which differ in the starting point used for the derivation of the algorithms, and which we will here refer to as semi-supervised clustering methods and semi-supervised classifiers. Semi-supervised clustering methods are based on traditional clustering methods that have been adapted to take additional external information into account. In contrast, semi-supervised classifiers are adapted versions of supervised classification methods that explicitly integrate unsupervised information into the training process: this is most commonly done through the transfer of decision boundaries into regions of low density (thus ensuring spatial continuity), for example in transductive support vector machines [13] or co-training [3]. The focus of this paper is on semi-supervised clustering methods.

2.2.1 Semi-supervised clustering methods

The adaptation of a clustering method for semi-supervision requires the integration of supervised information (either class labels or pairwise "must link" and "cannot link" constraints) into the clustering process. For this purpose, different components of the algorithm can be adapted, such as the initialization scheme, the distance function or the objective function. Alternatively, constraints reflecting the prior knowledge can be imposed on the set of possible clustering solutions, a strand of research also referred to as constrained clustering [9].

An adaptation of the initialization is probably the simplest approach and can, for example, be based on the use of the labeled data items to generate initial 'seed' clusters [2]. The distance function or the objective function of a clustering algorithm traditionally only reflect unsupervised information (that is, distances in data space), but can be adapted to consist of a linear or non-linear combination of supervised and unsupervised information components. Here, an adaptation of the distance function has the advantage that it can be 'plugged' into almost any clustering algorithm [11, 19]. In contrast, the optimization of a semi-supervised objective function will usually require the use of a general-purpose optimization method such as a genetic algorithm [8].

2.3 Motivation for the use of multiobjective optimization

To date, no thorough analysis of the advantages and disadvantages of the different existing methods of semi-supervised clustering is available, but a number of observations can be readily made. First of all, when integrating unsupervised and supervised information by means of a distance or an objective function, it is not usually clear what the best weighting between these components will be. It is possible that the weighting chosen may have a significant effect on the final outcome. Secondly, the use of hard constraints is also a choice that seems inflexible and may potentially cause constrained clustering methods to be sensitive to small annotation errors.

In this work, we argue that tackling the semi-supervised clustering problem within the framework of multiobjective optimization may provide a more flexible framework for the integration of both unsupervised and supervised components into the clustering process. Specifically, the use of Pareto optimization provides the means to avoid the need for hard constraints and for a fixed weighting between unsupervised and supervised objectives. Consequently, we would expect a multiobjective approach to semi-supervised clustering to perform more consistently across different data sets, and to show a higher robustness towards annotation errors.

3. MULTIOBJECTIVE SEMI-SUPERVISION

3.1 MOCK

In our previous work, we have described a multiobjective evolutionary algorithm (MOEA) for clustering, MOCK (Multiobjective clustering with automatic k-determination, [10]). The development of this algorithm was motivated by the difficulty of selecting a single clustering objective that performs robustly across a range of data with different properties. We have shown that this problem can be ameliorated through the use of multiple clustering objectives and through the identification of the trade-off solutions between these. In particular, MOCK has been compared to various single objective clustering methods and validation techniques, and results indicated a clear advantage to the multiobjective approach.

MOCK is based on the elitist multiobjective evolutionary algorithm, PESA-II, described in detail in [5].[1] It optimizes two clustering objectives, overall deviation and connectivity, which reflect two fundamentally different aspects of a good clustering solution: the global concept of compactness of clusters, and the more local one of connectedness of data points. The encoding employed is the locus-based adjacency scheme proposed in [17]. In this graph-based representation, each individual g consists of N genes $g_1, \ldots, g_N$, where N is the size of the clustered data set, and each gene g_i can take allele values j in the range $\{1, \ldots, N\}$. Thus, a value of j assigned to the ith gene, is then interpreted as a link between data items i and j: in the resulting clustering solution they will be in the same cluster. The decoding of this representation requires the identification of all connected components, and all items belonging to the same connected component are assigned to one cluster. This can be done in linear time. The operators used are standard uniform crossover, and a specialized initialization and mutation scheme. MOCK's initialization is based on minimum spanning trees (MSTs) and the k-means algorithm [16]. For a given data set, the complete MST is computed using Prim's algorithm. Individuals corresponding to different clustering solutions are then obtained by breaking up the MST, using a measure of interestingness of individual links and the partitionings prescribed by the k-means solutions. This has the effect of generating solutions with a range of cluster numbers that already provide a good and well-spread approximation to the Pareto front. MOCK's mutation operator allows data items to be linked to one of their L nearest neighbours only. Hence, $\forall i, g_i \in \{nn_{i1}, \ldots, nn_{iL}\}$, where nn_{il} denotes the lth nearest neighbour of data item i. This has the effect of significantly reducing the size of the search space.

MOCK generates clustering solutions that correspond to different trade-offs between the two clustering objectives and contain different numbers of clusters. In order to reduce the number of solutions to consider, an automated technique was developed, which selects good solutions from the resulting Pareto front, and, thus, simultaneously determines the number of clusters k in a data set. This method of solution selection is based on the shape of the Pareto front, specifically, it tries to determine 'knees' in the Pareto front that correspond to good solutions. A comparison to a null model, that is, random control data is used in order to correctly determine these knees. The use of the null model helps to abstract both from k-specific biases in the two objectives, and from biases introduced due to the shape of the underlying data manifold.

A positive side-effect of the multiobjective formulation of the clustering process is the ease of the integration of additional objectives into the existing framework. It should therefore be hoped that one possible approach to multiobjective semi-supervised clustering is a direct extension of MOCK through the integration of a third, supervised objective. This extension of MOCK is discussed in the following.

3.2 Extension of MOCK to semi-supervision

In order to extend MOCK to semi-supervision we integrate a third objective, which establishes the agreement between a given clustering solution and the prior knowledge. In our experimental setup, we assume that the prior knowledge is given in the form of class labels, and we can therefore use the Adjusted Rand Index (computed over the labeled data only) to obtain an objective assessment of the degree to which this prior knowledge has been preserved in a given clustering solution. Evidently, a modification of this third objective to account for different types of prior knowledge such as "must-link" and "cannot-link" constraints would be equally straightforward.

The Adjusted Rand Index is an external measure of clustering quality that is a generalization of the Rand Index. The Rand Indices are based on counting the number of pairwise co-assignments of data items. The Adjusted Rand Index additionally introduces a statistically induced normalization in order to yield values close to 0 for random partitions. This normalization removes the bias of the Rand Index with respect to different numbers of clusters, which is of particular importance in our application, as results across a range of cluster numbers are compared within the algo-

[1]The choice of this particular MOEA is motivated by our familiarity with the algorithm and is not believed to yield any particular advantage compared to other state-of-the-art MOEAs.

rithm. Using a representation based on contingency tables, the Adjusted Rand Index [12] is given as

$$R(U,V) = \frac{\sum_{lk}\binom{n_{lk}}{2} - \left[\sum_l \binom{n_{l\cdot}}{2} \cdot \sum_k \binom{n_{\cdot k}}{2}\right] / \binom{n}{2}}{\frac{1}{2}\left[\sum_l \binom{n_{l\cdot}}{2} + \sum_k \binom{n_{\cdot k}}{2}\right] - \left[\sum_l \binom{n_{l\cdot}}{2} \cdot \sum_k \binom{n_{\cdot k}}{2}\right] / \binom{n}{2}}, \quad (1)$$

where n_{lk} denotes the number of data items that have been assigned to both cluster l and cluster k. The Adjusted Rand Index returns values in the interval $[\sim 0, 1]$ and is to be maximized.

3.3 Alternative biobjective formulation of the semi-supervised clustering problem

In this section, we describe an alternative approach to semi-supervised clustering, which is based on two objectives only. This facilitates the visualization and exploration of the resulting Pareto fronts.

The basis for the implementation of a second, biobjective algorithm for semi-supervised clustering will be a single-objective evolutionary algorithm (EA) for clustering. Importantly, the encoding and operators developed for our multiobjective clustering algorithm can also be used in combination with clustering objectives other than connectivity and overall deviation. We can therefore maintain most of the components of MOCK described in Section 3.1, in particular the encoding, initialization, crossover and mutation operators. The only two changes necessary for the development of a single-objective EA for clustering is a modification of PESA-II's selection mechanism and the choice of a single clustering objective that is unbiased with respect to the number of clusters. For the single-objective case, selection from the archive is replaced by tournament selection of size two based on the single-objective fitness. The clustering objective chosen is the Silhouette Width, which is one of the most popular unsupervised validation techniques in the literature, and has also been used in previous work on semi-supervised clustering [11].

The Silhouette Width [18] for a partitioning is computed as the average Silhouette value over all data items. The Silhouette value for an individual data item i, which reflects the confidence in this particular cluster assignment, is

$$S(i) = \frac{b_i - a_i}{\max(b_i, a_i)}, \quad (2)$$

where a_i denotes the average distance between i and all data items in the same cluster and b_i denotes the average distance between i and all data items in the closest other cluster (which is defined as the one yielding the minimal b_i). The Silhouette Width returns values in the interval $[-1, 1]$ and is to be maximized.

The resulting single-objective EA can be extended to perform semi-supervision through the addition of the Adjusted Rand Index (on the labeled data) as a second objective. In this case, we simply reintroduce PESA II's original selection scheme.

4. RESEARCH QUESTIONS AND EXPERIMENTAL METHODOLOGY

In our experiments, we would like to explore the potential of a multiobjective formulation of the semi-supervised clustering problem. In particular, we are interested in the following questions.

1. On well-behaved synthetic data, can we observe significant performance differences between semi-supervised and entirely unsupervised or supervised approaches? By well-behaved, we here mean data that contain clear cluster structures, which are consistent with the real classes defined by the class labels.
2. On well-behaved synthetic data, can we observe significant performance differences between a genuine multiobjective semi-supervised approach and semi-supervised approaches based on a linear or non-linear combination of unsupervised and supervised objectives/distance functions?
3. On well-behaved synthetic data, how well do the different approaches cope with the introduction of noise into the class labels?
4. How do the algorithms compare on real data, which does not necessarily fulfil the assumptions made by semi-supervised classification, that is where unlabeled and labeled information may be inconsistent?

In order to address the above questions, we conduct comparisons between nine different methods on a range of synthetic and real data sets.

4.1 Contestant methods

Overall, we compare six semi-supervised, two supervised and one unsupervised method. These individual methods are described in the following.

1. Semi-supervised classification using MOCK (MOCK +semi). This version of the MOEA uses an additional third objective, which is the Adjusted Rand Index across the labeled data (as described in Section 3.2).
2. Unsupervised classification using MOCK (MOCK). This is our standard version of MOCK. Hence, two unsupervised clustering objectives are optimized across both labeled and unlabeled data.
3. Semi-supervised classification using MOCK and an adaptation of the distance matrix (MOCK+dist). The standard version of MOCK is used and, hence, two unsupervised clustering objectives are optimized across both labeled and unlabeled data. However, all distance values have been modified to take into account prior knowledge, as suggested in [11, 19]. Specifically, the distance between data items i and j is computed as $d = d_{\text{unsupervised}}(i,j) + d_{\text{supervised}}(i,j)$, where $d_{\text{unsupervised}}(i,j)$ is the normalized[2] Euclidean distance between i and j and
$$d_{\text{supervised}}(i,j) = \begin{cases} 0.5 & \text{if } i \text{ or } j \text{ is unlabeled,} \\ 0.0 & \text{if } i \text{ and } j \text{ have identical labels,} \\ 1.0 & \text{if } i \text{ and } j \text{ have different labels.} \end{cases}$$
4. Semi-supervised classification using multiobjective optimization (Semi). This version uses two objectives,

[2]In the initialization phase, all pairwise dissimilarity values are computed, and the maximum and minimum are identified. All pairwise distances are then scaled to lie within the interval $[0, 1]$.

which are the Silhouette Width across all data and the Adjusted Rand Index across the labeled data.

5. Semi-supervised classification using a non-linear combination of objectives (Non-linear). This version uses a single objective, which is the product of the Silhouette Width across all data and of the Adjusted Rand Index across the labeled data.
6. Semi-supervised classification using a linear combination of objectives (Linear). This version uses a single objective, which is a linear combination of the Silhouette Width across all data and of the Adjusted Rand Index across the labeled data. The Silhouette Width and the Adjusted Rand Index typically take values within similar ranges, and equal weighting of the two objectives is therefore used. This choice is in agreement with the suggestions in [8].
7. Unsupervised classification using the Silhouette Width (Unsupervised). This algorithm optimizes a single unsupervised clustering objective, which is the Silhouette Width across the labeled and unlabeled data.
8. Semi-supervised classification using a linear combination of distances (Unsupervised+dist). This algorithm optimizes a single unsupervised clustering objective, which is the Silhouette Width across the labeled and unlabeled data. The dissimilarity values have been modified as described for method 3 above.
9. Supervised classification (Supervised). A five-nearest neighbour classifier based on the labeled data[3] is used [7].

The parameter settings used are identical to those given in [10], apart from the total number of generations, which we set to 1000.

4.2 Data sets

In order to analyze the different algorithms on data sets, in which cluster structures are clearly discernible and in which the class labels are consistent with the structures present, we use a collection of synthetic data sets. These are obtained using a data generator for multivariate Gaussian clusters whose data sets have been shown to be hard to solve for a variety of different algorithms [10]. The generator is applied to produce a number of small data sets with $k \in \{2, 4, 10\}$ and $d \in \{2, 10\}$ (individual cluster sizes are uniformly distributed within the interval $\{10, \ldots, 50\}$). We produce ten instances of each type. In our experiments, all ten data sets that are of dimensionality d and contain k clusters are then grouped and referred to as the group of data dd-kc. Hence, in total we obtain 6 different groups of data sets, which consist of 10 individual instances each. Some sample data sets are shown in Figure 2.

In addition, we use real data sets taken from the Machine Learning Repository [6]. The Iris, Wine, Zoo and Dermatology data sets are used, whose dimensionalities range from 4 (Iris) to 34 (Dermatology), whose number of clusters range from 3 (Iris and Wine) to 7 (Zoo), and whose sizes range from 101 (Zoo) to 366 (Dermatology). The cluster structures in most of these real data sets are not clearly discernible, and the degree of consistency between the structures present and the class labels is not clear. The variable ranges within this data can vary drastically, and we therefore use normalization to a mean of 0 and a standard deviation of 1.

[3]Evidently, this means that only a very small amount of training data (here, 5 items per class) are used, and the supervised method can not be expected, therefore, to yield the same performance observed in the literature when training on all available data. This scenario of limited available training data is where semi-supervised approaches would be expected to be superior to supervised approaches.

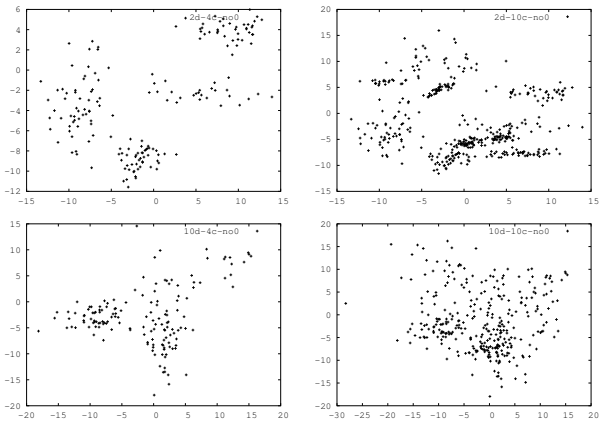

Figure 2: Some two-dimensional projections of the synthetic data data sets used in our study. Evidently, the cluster structures in these data are discernible. On the other hand, the data sets are difficult enough to reveal distinct performance differences between the different methods.

For both the synthetic and the real data sets, the true classification, that is the class labels for all data items are known, and we can therefore objectively assess the quality of a given clustering result. During the classification process, we only use a fraction of the class labels available, in order to simulate the availability of limited prior class knowledge. Hence, the data is divided into unlabeled and labeled data, which correspond to training and testing data respectively. Consistently with the principles of transductive inference, both the unlabeled and the labeled data are used during the classification process.

In total, four different groups of experimental data with varying degrees of annotation and noise levels (in the class labels of the labeled data) are created.

1. Synthetic data 1: The first group of experiments considers the performance of the algorithms on sparsely labeled data without noise. Specifically, exactly 5 items (selected uniformly at random) of each class are labeled, and all remaining data items are treated as unlabeled data.
2. Synthetic data 2: The second group of experiments considers the performance of the algorithms if noise is introduced into the experiments. Specifically, exactly 5 labeled items of each class are labeled, but one out of the five is assigned an incorrect class label. All remaining data items are treated as unlabeled data.
3. Synthetic data 3: The third group of experiments considers the performance of the algorithms on more general data with an arbitrary number of class labels and an arbitrary degree of noise. Specifically, the percentage of labeled data items for every cluster is deter-

mined uniformly at random and may vary from 0 to 100 per cent. The percentage of incorrect class labels within this group is also determined uniformly at random and may vary from 0 to 50 per cent.

4. Real data: The fourth group of experiments considers the performance of the algorithms on sparsely labeled real data. Specifically, exactly 5 items of each class are labeled (selected uniformly at random), and all remaining data items are treated as unlabeled data. Due to the real nature of this data, the degree of noise present in the data is unknown.

4.3 Performance evaluation

In order to evaluate the quality of all solutions in the Pareto front, the Adjusted Rand Index is calculated for the unlabeled (testing) data. The use of the unlabeled data only ensures that the results obtained by the unsupervised, semi-supervised and supervised algorithms can be justly compared. We then analyze the quality of the best solution identified by the different algorithm, that is we use external knowledge in order to select the best solution present in the Pareto front. Evidently, this would not be possible in a real application (where the correct solution is not known), and the development of an internal method of selecting the best solution would be preferable. We will further address this important issue in Section 6.

In the subsequent analysis, we partition the algorithms into two different groups based on the clustering objective used in the algorithms. Group A comprises the three algorithms based on MOCK, that is those that optimize connectivity and overall deviation. Group B comprises the remaining five genetic algorithms, which all optimize the Silhouette Width, as well as the nearest neighbour classifier. This grouping is motivated by the fact that only algorithms within these groups can be justly compared with regard to the impact of semi-supervision.

A Wilcoxon Signed Rank test is applied to each pair of algorithms' results within a group. This is a nonparametric test for differences between two paired (or matched) samples, as described in [4]. A paired samples test is used because data sets within a group may be heterogeneous, e.g. the group 2d-4c is made up of 10 data sets, and, in addition, different fractions of labeled data and noise are used in the third group of experimental data. The two-tailed significance level $\alpha = 0.01$ is used. With the Bonferroni correction, this means that results have an overall significance of $\alpha_{\text{overall}} = 0.05$ or better. Those, and only those, algorithms that are not significantly worse than any other are deemed to be best performers.

5. RESULTS

The results of our experiments (averages over 21 runs) are summarized in Table 1 to Table 4.

5.1 Analysis of the results in Group B

The results obtained for the algorithms in Group B provide tentative answers for all of the questions raised in Section 4. Firstly, the results in Table 1 indicate that, on sparsely labeled data, semi-supervised approaches can indeed have significant performance advantages compared to purely unsupervised and supervised approaches. In particular, the semi-supervised approaches based on Pareto optimization and a linear or non-linear combination of the objectives all outperform the unsupervised and supervised methods on these data. Of these three, the method based on Pareto optimization emerges as the strongest performer. Secondly, Table 2 and Table 3 demonstrate that the algorithms differ in their robustness towards increasing degrees of noise. While the performance of the multiobjective approach remains nearly unchanged, despite increasing noise levels, the performances of the other semi-supervised and supervised approaches break down significantly. In order to further analyze these differences between the algorithms, we have run additional experiments studying the algorithms' performance as a function of the percentage of labels and the noise level. These results confirm the high robustness of the Pareto-based approach. A representative result for a degree of labeling of 20 per cent is shown in Figure 3.

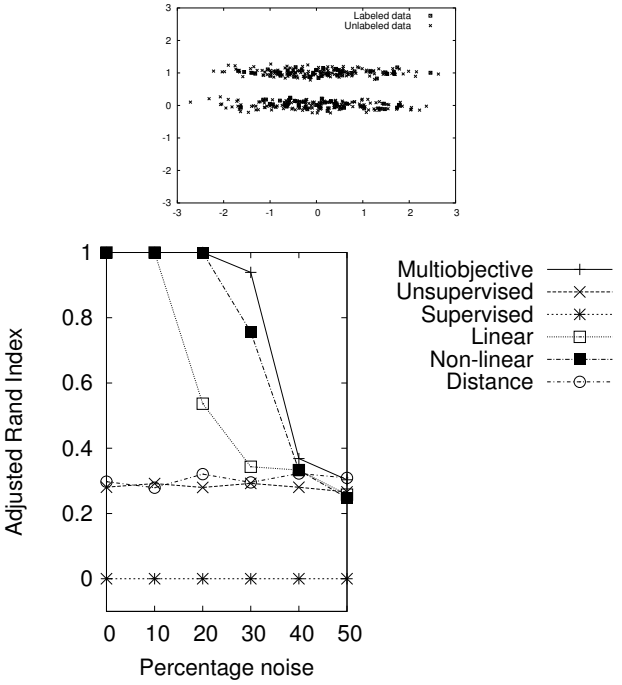

Figure 3: Robustness of the algorithms in Group B towards noise (all algorithms in Group A consistently score 1.0 on this data set). Top: the 'Long' data set. Bottom: results obtained on the 'Long' data set as a function of the class label noise level when 20 per cent of the data are labeled (averages over 21 runs).

The overall performance advantage of a Pareto-based optimization approach is also confirmed by the results obtained on the real data.

5.2 Analysis of the results in Group A

The performance differences observed in Group A are less pronounced, which is due to the already strong performance of the underlying unsupervised method, MOCK. However, a small but consistent performance advantage for the semi-supervised approach remains, which is particularly significant on two of the four data sets in Table 4. These results may indicate a genuine degree of conflict between unsupervised and supervised information in these real-world data sets.

Table 1: Results for data with 5 labeled items per cluster. The left three and right six results are compared separately, and the statistically best performers are identified in bold font. See Section 4.3 for information on the statistical testing procedure.

	Group A			Group B					
Data set	Mock+semi	MOCK	MOCK+dist	Semi	Non-linear	Linear	Unsupervised	Unsupervised+dist	Supervised
2d-2c	0.991884	0.991884	**0.999903**	**0.920614**	**0.919167**	**0.918757**	**0.918561**	0.896844	0.900506
2d-4c	**0.942098**	0.905061	0.908425	**0.847385**	0.819923	0.82711	0.682194	0.69447	0.809218
2d-10c	**0.899483**	0.874762	0.874235	**0.853507**	0.831026	0.829606	0.690712	0.717952	0.802053
10d-2c	**1**	**1**	**1**	**0.994937**	**0.994937**	**0.994937**	**0.994937**	**0.994937**	0.943913
10d-4c	**0.985728**	**0.986181**	**0.986768**	**0.965785**	0.959128	0.956302	0.862566	0.91597	0.910616
10d-10c	**0.955528**	0.950558	0.947395	**0.92053**	0.912364	0.912763	0.880128	0.899637	0.805448

Table 2: Results for data with 5 labeled items per cluster, including one mis-annotation. The left three and right six results are compared separately, and the statistically best performers are identified in bold font. See Section 4.3 for information on the statistical testing procedure.

	Group A			Group B					
Data set	Mock+semi	MOCK	MOCK+dist	Semi	Non-linear	Linear	Unsupervised	Unsupervised+dist	Supervised
2d-2c	**0.993057**	0.991884	0.991884	**0.925497**	0.82724	0.84546	0.918712	0.916358	0.869614
2d-4c	**0.942196**	0.904442	0.907416	**0.832011**	0.791719	0.791319	0.665082	0.662503	0.77426
2d-10c	**0.897458**	0.873571	0.872091	**0.848273**	0.818969	0.818266	0.684205	0.697709	0.801014
10d-2c	**1**	**1**	**1**	**0.994937**	0.836226	0.759519	**0.994937**	**0.994937**	0.862411
10d-4c	**0.986456**	**0.985211**	**0.985843**	**0.966461**	0.949985	0.926525	0.879349	0.899708	0.841178
10d-10c	**0.955449**	0.947991	0.947531	**0.920454**	0.905918	0.893194	0.880739	0.889057	0.727699

Table 3: Results for data with an arbitrary percentage of labels and up to 50 percent of noise in the labels. The left three and right six results are compared separately, and the statistically best performers are identified in bold font. See Section 4.3 for information on the statistical testing procedure.

	Group A			Group B					
Data set	Mock+semi	MOCK	MOCK+dist	Semi	Non-linear	Linear	Unsupervised	Unsupervised+dist	Supervised
2d-2c	**0.991822**	0.990982	0.990754	**0.920693**	0.765729	0.846631	0.893491	0.879105	0.478551
2d-4c	**0.924401**	0.898159	0.896986	**0.850144**	0.783256	0.782977	0.681509	0.711387	0.628065
2d-10c	**0.880754**	0.865354	0.863996	**0.831541**	0.774617	0.778177	0.676895	0.637842	0.649413
10d-2c	**0.998264**	**0.998264**	**0.998264**	**0.996784**	0.847696	0.88651	0.996075	0.954799	0.447426
10d-4c	**0.983065**	**0.983969**	**0.986374**	**0.957709**	0.934209	0.931787	0.852166	0.813769	0.622255
10d-10c	**0.958176**	**0.957562**	**0.955065**	**0.929012**	0.914031	0.912206	0.898567	0.767229	0.587158

Table 4: Results for data with 5 labeled items per cluster. The left three and right six results are compared separately, and the statistically best performers are identified in bold font. See Section 4.3 for information on the statistical testing procedure.

	Group A			Group B					
Data set	Mock+semi	MOCK	MOCK+dist	Semi	Non-linear	Linear	Unsupervised	Unsupervised+dist	Supervised
iris	**0.850532**	0.704117	0.700442	**0.756913**	0.70355	0.683191	0.567742	0.567742	**0.7577**
wine	**0.793531**	**0.775236**	**0.788655**	0.792217	**0.807097**	**0.810813**	0.486009	0.476542	0.447076
derma	**0.898172**	0.853476	0.850444	**0.86911**	0.848901	0.82834	0.592587	0.594184	0.766839
zoo	**0.985262**	**0.985174**	**0.985174**	**0.891337**	0.894904	0.894977	0.900344	0.895436	0.885687

6. DISCUSSION AND CONCLUSIONS

The experiments presented in this paper indicate several advantages of a multiobjective approach to semi-supervised clustering, and, more generally, those of a semi-supervised approach to classification. We find, in particular, that the multiobjective approach shows a more consistent performance across different data sets, and is more robust to noise.

On the other hand, one limitation of our analysis has been the use of external knowledge to choose the best solution from the Pareto fronts in the case of the multiobjective approaches. Here, we were interested in the algorithms' performance at *generating* high quality solutions, and we believe an advantage has been shown for the multiobjective methods in this respect. However, in practice, the selection of the best solution from the Pareto front may pose difficulties, and the peak performance obtained in these experiments may therefore be difficult to reach in reality. Nonetheless, in our previous work on multiobjective clustering, we have successfully developed an automated and unsupervised scheme for the selection of good solutions from the Pareto front, and in future work we hope to develop similar approaches for the Pareto fronts obtained by multiobjective semi-supervised clustering.

It should also be noted that the MOEA used in our work is not necessarily able to genuinely explore all of the trade-offs between unsupervised and supervised information. This is due to the encoding and the operators used, which (for the sake of efficiency) need to integrate domain knowledge

and are, in particular, based on the assumption of coherence in data space. Solutions that strongly violate this assumption can, therefore, not necessarily be reached. While it may be theoretically interesting to explore the entire range of the Pareto front, we do not consider this limitation a disadvantage innate to our approach. In fact, all existing semi-supervised clustering algorithms are based on similar assumptions, and the use of this assumption seems perfectly justified: as mentioned previously, the idea of coherence (or consistency and complementarity between supervised and unsupervised information) is fundamental in semi-supervision and is the main ground for its potential advantages — consequently, semi-supervision makes little sense in the complete absence of coherence. Admittedly, a high degree of coherence may not be immediately present in many real data sets encountered in practical data-mining scenarios. However, this does not imply that an exploration of more extreme trade-offs would be useful, but rather that a transformation of the feature space will be necessary to obtain coherent data and find interpretable results. For this reason, the integration of semi-supervision with feature selection or with distance learning is an important area of research and will be a logical extension of the work presented in this paper.

Acknowledgments

JH acknowledges support of a doctoral scholarship from the German Academic Exchange Service (DAAD) and the Gottlieb Daimler- and Karl Benz-Foundation, Germany. JK is supported by a David Phillips Fellowship from the Biotechnology and Biological Sciences Research Council (BBSRC), UK.

7. REFERENCES

[1] E. Bair and R. Tibshirani. Semi-supervised methods to predict patient survival from gene expression data. *PLoS Biol*, 2(4):0511–0521, 2004.

[2] S. Basu, A. Banerjee, and R. Mooney. Semi-supervised clustering by seeding. In *Proceedings of the 19th International Conference on Machine Learning*, pages 19–26. Morgan Kaufmann Publishers, San Francisco, CA, 2002.

[3] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *Proceedings of the Conference on Computational Learning Theory*. ACM Press, New York, NY, 1998.

[4] W. J. Conover. *Practical Nonparametric Statistics, second edition*. John Wiley & Sons, New York, NY, 1980.

[5] D. W. Corne, N. R. Jerram, J. D. Knowles, and M. J. Oates. PESA-II: Region-based selection in evolutionary multiobjective optimization. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 283–290. Morgan Kaufmann Publishers, San Francisco, CA, 2001.

[6] C. L. Blake D. J. Newman, S. Hettich and C. J. Merz. UCI repository of machine learning databases, 1998.

[7] B. V. Dasarathy. *Nearest neighbour (NN) norms: NN pattern classification techniques*. IEEE Computer Society, Washington, DC, 1991.

[8] A. Demiriz, K. P. Bennett, and M. J. Embrechts. A genetic algorithm approach for semi-supervised clustering. *Smart Engineering System Design*, 4:21–30, 2002.

[9] A. D. Gordon. A survey of constrained classification. *Computational Statistics & Data Analysis*, 21:17–29, 1996.

[10] J. Handl and J. Knowles. Improvements to the scalability of multiobjective clustering. In *Proceedings of the 2005 IEEE Congress on Evolutionary Computation*, pages 2372–2379. IEEE Press, Anaheim, CA, 2005.

[11] D. Hanisch, A. Zien, R. Zimmer, and T. Lengauer. Co-clustering of biological networks and gene expression data. *Bioinformatics*, 18(90001):145S–154, 2002.

[12] A. Hubert. Comparing partitions. *Journal of Classification*, 2:193–198, 1985.

[13] T. Joachims. Transductive inference for text classification using support vector machines. In *Proceedings of ICML-99, 16th International Conference on Machine Learning*, pages 200–209. Morgan Kaufmann Publishers, San Francisco, CA, 1999.

[14] M.-A. Krogel and T. Scheffer. Effectiveness of information extraction, multi-relational, and semi-supervised learning for predicting functional properties of genes. In *Proceedings of the Third IEEE International Conference on Data Mining*, pages 569–573. IEEE Press, New York, NY, 2003.

[15] T. Li, S. Zhu, Q. Li, and M. Ogihara. Gene functional classification by semi-supervised learning from heterogeneous data. In *Proceedings of the Symposium on Applied Computing*, pages 78–82. ACM Press, New York, NY, 2003.

[16] L. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, pages 281–297. University of California Press, Berkeley, CA, 1967.

[17] Y.-J. Park and M.-S. Song. A genetic algorithm for clustering problems. In *Proceedings of the Third Annual Conference on Genetic Programming*, pages 568–575. Morgan Kaufmann Publishers, San Francisco, CA, 1998.

[18] P. J. Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987.

[19] N. Speer, C. Spieth, and A. Zell. A memetic co-clustering algorithm for gene expression profiles and biological annotation. In *Proceedings of the Congress on Evolutionary Computation*, pages 1631–1638. IEEE Press, Anaheim, CA, 2004.

[20] V. Vapnik. *Statistical Learning Theory*. John Wiley and Sons, New York, NY, 1998.

[21] J. Weston, C. Leslie, D. Zhou, A. Elisseeff, and W. Noble. Semi-supervised protein classification using cluster kernels. In *Proceedings of the International Conference on Neural Information Processing Systems 2003*. The Internet, 2003.