

Semi-supervised feature selection via multiobjective optimization

Julia Handl and Joshua Knowles

Abstract—In previous work, we have shown that both unsupervised feature selection and the semi-supervised clustering problem can be usefully formulated as multiobjective optimization problems. In this paper, we discuss the logical extension of this prior work to cover the problem of semi-supervised feature selection. Our extensive experimental results provide evidence for the advantages of semi-supervised feature selection when both labelled and unlabelled data are available. Moreover, the particular effectiveness of a Pareto-based optimization approach can also be seen.

I. INTRODUCTION

By the term, ‘clustering’, one usually refers to the identification of homogeneous groups of data items within a data set and, more specifically, to the identification of a set of ‘clusters’, which group highly similar items and dissociate dissimilar ones. As clustering works without the use of any type of training data, it is particularly useful in the absence of prior knowledge about the patterns present in a given data set, and it is, therefore, one of the standard tools for exploratory data analysis. The term clustering is frequently employed interchangeably with the term unsupervised classification, but, in fact, clustering algorithms are only one of the techniques comprising unsupervised classification, which also includes unsupervised methods for dimensionality reduction, feature subset selection and general transformations of the feature space.

The common factor in all methods of unsupervised classification is that they eschew external guidance, relying instead completely on the patterns intrinsic to the data in the feature space. This means that these methods can only be expected to work if a data set contains clear patterns, and must fail if the significant patterns are masked by noise or experimental artefacts. This limitation is often encountered when attempting to employ clustering methods in practice, which is usually done using a standard distance function (such as the Euclidean or the Correlation distance) operating within the full, original feature space. Clusters intrinsic to smaller feature subspaces of the data are likely to be overlooked when using such global distance measurements.

In order to obtain interesting results in practice, it is, therefore, of importance to develop methods that are able to reduce the dimensionality of the original feature space and to explore patterns in low-dimensional feature subspaces. Most existing methods for feature selection are supervised, that is, they use labelled training data in order to identify those features contributing to a good discrimination between classes. However, these methods usually require the presence

of sufficiently large sets of training patterns, which are often not available in an unsupervised classification scenario. Recent research has therefore investigated methods of feature selection that are applicable in the presence of no or very little training data, that is, methods of unsupervised and semi-supervised feature selection. Unsupervised feature subset selection aims to identify subspaces containing clear cluster structures where the quality of the partitioning is evaluated using only internal techniques of cluster validation, that is, measures that assess the degree of structure captured by a given partitioning. Methods of semi-supervised feature selection additionally integrate a small percentage of supervised (external) information into this search process. Such ‘sparse’ prior knowledge is available in many real world applications (for example, in the analysis of gene expression data, where the function of a small number of genes is usually known), and, whilst often not sufficient to conduct an entirely supervised analysis, may help to direct the search towards subspaces of high interest.

In our previous work, we have shown the advantages of a multiobjective formulation of a number of problems in classification, including clustering [11], unsupervised feature selection [12] and semi-supervised clustering [13]. In this paper, we integrate aspects from some of this prior work and devise a semi-supervised algorithm for feature selection. The remainder of this paper is structured as follows. Section II reviews previous research on semi-supervised feature selection. Section III recalls our own work on unsupervised feature selection and semi-supervised clustering, and describes how this work is combined and extended to obtain an algorithm for semi-supervised feature selection. In Section IV, we set out the main research questions investigated and explain our corresponding experimental setup. Results of our experiments are presented in Section V and, finally, Section VI concludes.

II. RELATED WORK

A. Semi-supervised classification

The input to a semi-supervised classification problem is, typically, a data set consisting of unlabelled and labelled data, with the amount of labelled data being relatively limited. Due to this sparseness of the labelled ‘training data’, the supervised classification problem on this data will, usually, be underdetermined, and the models resulting from an entirely supervised analysis may therefore be meaningless. On the other hand, an entirely unsupervised analysis may produce a partitioning not consistent with the class labels available, and may therefore be of little interest to the user. Semi-supervised classification aims to find a solution to these classification problems that is consistent both with the data

Julia Handl and Joshua Knowles are with the Manchester Interdisciplinary Biocentre, University of Manchester, Manchester, UK (email: j.handl@postgrad.manchester.ac.uk, j.knowles@manchester.ac.uk).

distribution (internal knowledge) and prior information about class memberships or related constraints (external knowledge). Different approaches to the integration of these two sources of information exist, and these fundamentally differ in the underlying algorithms and their bias to one or the other of the two types of information. One common strand of research uses established supervised classifiers, such as support vector machines: these classifiers are trained on the labelled data, but decision boundaries between classes are ‘shifted’ into areas of low densities, as measured across the unlabelled data [3], [16]. An alternative approach is the use of established clustering methods, such as k -means or agglomerative algorithms [1], [14]. Evidently, these are principally guided by the unlabelled data, but the labelled data may be used to bias the search towards clusters consistent with the labelled data. In this paper, we are chiefly interested in the latter approach, also termed semi-supervised clustering, and we will now briefly discuss previous work in this respect.

B. Semi-supervised clustering methods

The adaptation of a clustering method for semi-supervision requires the integration of external information 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 [10].

An adaptation of the initialization is probably the simplest approach and can, for example, be based on the use of the labelled data items to generate initial ‘seed’ clusters [1]. The distance function or the objective function of a clustering algorithm traditionally reflect unsupervised information only (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 [14], [20]. 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 [9].

C. Semi-supervised metric learning and feature selection

One of the fundamental assumptions of semi-supervised classification is the complementarity and consistency between internal and external information. If the assumption of consistency is violated, that is if internal and external information are violently contradicting, the application of semi-supervised methods makes little sense, as the methods cannot be expected to profit from the combined use of the two information sources.

However, unfortunately, the assumption of consistency is violated in many real-world scenarios, that is, it is often not fulfilled in the original full feature space. Consequently, the successful use of a semi-supervised clustering technique may require the identification of a transformed feature space that

is better suited to reflect the relationships defined by the class labels provided.

Recently, a number of papers have investigated semi-supervised approaches aimed at a transformation of the original feature space by means of metric learning or feature selection [2], [4], [22], [23]. Metric learning aims to identify a distance function that provides distance values consistent with the class labels provided. Usually, the space of distance functions considered are standard distance functions (such as the Euclidean distance or Mahalanobis distance) across weighted feature spaces [2], [4], [22]. Feature subset selection can be seen as a subset of metric learning, where only discrete (0/1) weights are permitted [23].

D. Motivation for the use of multiobjective optimization

None of the existing methods for semi-supervised metric learning and feature selection fully exploit the potential of using both external and internal information simultaneously. Some of these methods integrate internal and external information only in some steps of the algorithm, for example in the method proposed in [23], the best feature space is identified in a purely supervised way. Others use an adapted clustering criterion that consists of a fixed linear combination of external and internal objective components [2], which may be inflexible and may lead to sub-optimal results. In this paper, we therefore aim to explore whether advantages can be gained through the simultaneous optimization of both internal and external components within the framework of Pareto optimization.

III. SEMI-SUPERVISED FEATURE SELECTION

In previous work, we have developed algorithms for unsupervised feature selection and semi-supervised clustering. In this section, we will review the algorithms developed for these tasks and consider how aspects of the two can be integrated to tackle semi-supervised feature selection.

A. Previous work on semi-supervision

In [13], we have described a multiobjective evolutionary approach to semi-supervised clustering. In this algorithm, the concept of semi-supervision is implemented by optimizing separate objectives related to the performance with respect to internal and external information. Specifically, the algorithm works through the optimization of an internal cluster validation technique, the Silhouette Width [19], combined with the optimization of an external validation technique, the Adjusted Rand Index [15]. The Silhouette Width is computed across both labelled and unlabelled data, whereas the Adjusted Rand Index can be computed for the labelled data only.

The optimization algorithm used is an existing multiobjective evolutionary algorithm (MOEA) from the literature, PESA-II [6]. In order to obtain good scalability to large data sets, a specialized encoding and specialized operators are used. In particular, both the encoding and the mutation operator make use of nearest neighbour lists to restrict the size of the search space.

The resulting method was compared to unsupervised and supervised approaches, as well as alternative semi-supervised algorithms based on the integration of unsupervised and supervised information within an individual objective or through the distance function. The experimental results indicate a significant advantage of the multiobjective semi-supervised approach.

Performing feature selection requires an efficient search through different feature subspaces and the identification of appropriate clustering solutions for every subspace. A straightforward use of our existing semi-supervised clustering method is problematic, as its application would require the re-computation of the nearest-neighbour lists for every subspace, and the subsequent application of the full algorithm. This would be computationally prohibitive even for small data sets. We therefore need to investigate more efficient methods of assessing the quality of a given feature space.

B. Previous work on unsupervised feature selection

We have addressed the problem of unsupervised feature selection [12] and, in this work, the k -means algorithm [17] was employed to obtain an efficient cluster generator for the assessment of individual feature subspaces. The k -means algorithm was chosen due to its linear runtime and the ease of interpretation of the resulting clustering solutions.

As discussed in detail in [12], a multiobjective formulation of unsupervised feature selection was found to be an effective method of dealing with the inherent cardinality-bias of internal techniques for cluster validation. We therefore proposed the simultaneous maximization of the Silhouette Width (which is biased towards small feature cardinalities) and the maximization of the feature cardinality. PESA-II is used to evolve feature sets as well as the corresponding number of clusters and the k -means algorithm is used to generate a partitioning for a given individual. A good performance of this algorithm was demonstrated in a comparison to alternative choices of objectives and optimizers.

C. Implementation of semi-supervised feature selection

Using aspects of both of the algorithms described above, we will now describe an algorithm for semi-supervised feature selection. It is primarily based on the method for unsupervised feature selection, which can be extended to account for semi-supervision by means of the following changes.

- 1) The addition of a third objective taking into account the consistency with the given constraints or class labels.
- 2) The adaptation of the initialization scheme to obtain a good initialization along the third dimension of the Pareto front.
- 3) The development of a new method for solution selection.

The architecture of the resulting algorithm is illustrated in Figure 1. We will now briefly outline all of the main components.

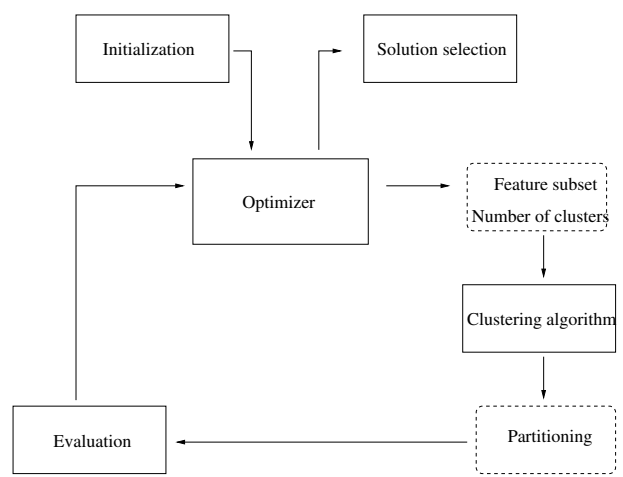


Fig. 1. The main components of our algorithm for multiobjective semi-supervised feature selection. After an initialization phase, the main loop of the algorithm is started. A search algorithm/optimizer constructs candidate solutions that specify a feature subset and the number of clusters. Each solution then serves as the input to a clustering algorithm, which, in the given feature subspace, partitions the data into the number of clusters specified. The resulting partitioning is evaluated and the resulting objective values are fed back to the optimizer. The main cycle is iterated for a pre-specified number of iterations. The final output of the algorithm is the set of Pareto optimal solutions. Ideally, the algorithm would include an additional module for solution selection, which selects good solutions from the Pareto front.

1) *PESA-II*: The optimizer used is the elitist MOEA, PESA-II, described in detail in [6].¹ Briefly, PESA-II updates, at each generation, a current set of non-dominated solutions stored in an external population (of non-fixed but limited size), and uses this to build an internal population of fixed size to undergo reproduction and variation. PESA-II uses a selection policy designed to give equal reproduction opportunities to all regions of the current non-dominated front; thus in our application, it should provide a diverse set of solutions trading off the three different objectives. No critical parameters are associated with this ‘niched’ selection policy, as it uses an adaptive range equalization and normalization of the objectives.

2) *Encoding and variation operators*: The application of PESA-II to feature selection requires the choice of an appropriate encoding and operators. Due to the use of k -means, there are two components of a solution that need to be coded for: the actual feature subset, and the number of clusters. A simple binary encoding is used to select/deselect features: the genome comprises one bit for every feature, with a value of 1 indicating the activation of a feature and a value of 0 indicating its deactivation. The variation operators applied to this part of the genome are uniform crossover (with a standard crossover probability of 0.7) and bit-flip mutation (with a mutation probability of $\frac{1}{d}$ where d is the total number of features available). Four-bit Gray coding is used to encode the number of clusters, constrained to the

¹The 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.

range $k \in \{2, \dots, 17\}$. The variation operator applied to this part of the genome is bit-flip mutation (with a mutation probability of $\frac{1}{4}$).

3) *Clustering algorithm*: In certain respects, the ideal choice for the clustering algorithm would be a powerful clustering method that is capable of detecting clusters of very different types (such as clusters of arbitrary shape, overlapping clusters or unequally sized clusters). Unfortunately, the clustering algorithm needs to be run for every single evaluation and the use of an algorithm with high computational complexity is therefore undesirable. In our experiments, we decide on the use of k -means, which seeks compact clusters, but whose time complexity is only linear in the number of data items.

The k -means algorithm starts from a random partitioning of the data into k clusters (where k is an input parameter). It repeatedly (i) computes the current cluster centres (that is, the average vector of each cluster in data space) and (ii) reassigns each data item to the cluster whose centre is closest to it. It terminates when no more reassignments take place. By this means, the intra-cluster variance, that is, the sum of squares of the differences between data items and their associated cluster centres, is locally minimized.

Our implementation of the k -means algorithm is based on the batch version of k -means, that is, cluster centres are only recomputed after the reassignment of all data items. Random initialization (which is known to be an effective initialization method [18]) is used.

4) *Objective functions*: The resulting clustering solutions are evaluated using three different objectives.

The first of these takes into account the quality of the partitioning with respect to internal information. The internal validation technique chosen for this purpose 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 [14]. The Silhouette Width [19] 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

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

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 second objective used is the feature cardinality, which needs to be maximized in order to counterbalance the cardinality bias of the Silhouette Width.

The third objective takes into account the preservation of the external knowledge available for a data set. For this purpose, the Adjusted Rand Index is computed over the labelled data only. The Adjusted Rand Index is an external measure of clustering quality and 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 algorithm. Using a representation based on contingency tables, the Adjusted Rand Index [15] is given as

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

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.

5) *Constraints*: The size of the full search space of the feature selection problem grows exponentially with the numbers of features. Yet, in most applications, researchers are predominantly interested in finding partitionings in feature subspaces that involve a relatively small number of variables only. In order to allow for an efficient search by the algorithm through these low-dimensional subspaces, a constraint on the maximum cardinality of the feature subspaces considered is imposed, which reduces the size of the search space to $O(d^{d_{max}})$. In all of the experiments presented in this paper, this constraint is set to $d_{max} = \min(20, d)$.

D. Initialization

A heuristic initialization scheme is implemented that aims to seed the optimization method with good initial feature sets. This initialization phase works as follows. First, all possible feature sets of cardinality 1 are constructed. All of these singleton feature sets are evaluated for a fixed number of $k = 2$ clusters. The Silhouette Width and the Adjusted Rand Index of each singleton feature set are computed, and, for each of the two objectives, a list is created, which sorts these feature sets in decreasing order of the objective value. The initial population of size $2 \times d_{max}$ (recall that d_{max} is the constraint on the maximum cardinality of the feature space) is then generated as follows: for $i = 1, \dots, d_{max}$, the i th solution within this population is constructed by combining the i features with the highest individual scores under the Silhouette Width (again, a fixed number of $k = 2$ clusters are used). The same process is repeated using the scores obtained for the Adjusted Rand Index.

E. Solution selection

Assuming the absence of noise in the external information, we can assume that the solution performing best under the supervised objective will be the best solution. This provides us with a mechanism to select a single solution from the Pareto front.

The two main questions addressed in this paper relate to the advantages afforded by the use of semi-supervision, and those afforded by means of a multiobjective Pareto-based approach. Regarding the first issue, we are interested in identifying whether the integration of sparse prior knowledge can indeed improve upon the performance of an entirely unsupervised method for feature selection. Regarding the second issue, it is unclear whether the explicit optimization of objectives related to internal and external information by means of a multiobjective Pareto-based approach will produce results superior to those obtained using an algorithm based on the optimization of a linear or non-linear combination of these two objectives.

A. Contestant methods

In order to address these questions, we perform comparisons of five alternative algorithms. These different methods are all based on the MOEA described above, and only differ in the number and the choice of objectives.

- 1) Semi-supervised feature selection. This version of the MOEA uses three objectives, (i) the Silhouette Width on all data (unsupervised objective), (ii) the Adjusted Rand Index on the labelled data (supervised objective) and (iii) the feature cardinality, which is maximized.
- 2) Unsupervised feature selection. This version of the MOEA uses two objectives only, (i) the Silhouette Width on all data (unsupervised objective) and (ii) the feature cardinality, which is maximized.
- 3) Supervised feature selection. This version of the MOEA uses two objectives only, (i) the Adjusted Rand Index on the labelled data² (supervised objective) and (ii) the feature cardinality, which is minimized. Minimization of the number of features is necessary in this context to avoid overtraining and our choice in agreement with previous research on the multiobjective optimization of supervised classifier performance [8].
- 4) Non-linear combination. This version of the MOEA uses two objectives only, (i) the product of the Silhouette Width on all data and the Adjusted Rand Index on the labelled data (semi-supervised objective) and (ii) the feature cardinality, which is maximized.
- 5) Linear combination. This version of the MOEA uses two objectives only, (i) a linear combination of the Silhouette Width on all data and the Adjusted Rand Index on the labelled data (semi-supervised objective) and (ii) the feature cardinality, which is maximized. 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 also in agreement with the setup used in [20].

²Evidently, 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.

B. Parameter settings

The parameter settings used in our experiments are summarized in Table I. Apart from the number of initial solutions and the total number of generations, these are identical for all five MOEAs.

TABLE I
PARAMETER SETTINGS FOR THE FIVE MOEAS.

<i>Parameter</i>	<i>setting</i>
Maximum feature cardinality d_{max}	20
Number of generations	2000 or $2000 + \lceil \frac{d+d_{max}}{ipsize} \rceil$
External population size	1000
Grid resolution per dimension	10
Initial population size $ipsize$	$2 \times d_{max}$ or d_{max}
Internal population size	10
Mutation rate on features	$\frac{1}{d}$
Mutation rate on cluster number	$\frac{1}{J}$
Range of cluster numbers explored	$k \in \{2, \dots, 17\}$
Recombination on features	Uniform crossover
Recombination on cluster number	None
Recombination rate	0.7
Constraints	$d \leq d_{max} \wedge k \in [2, 17]$

The number of initial solutions used in the three-objective algorithm is $2 \times d_{max}$, which is twice the number of the initial solutions used by the bi-objective algorithms. The number of iterations for the three-objective algorithm is set to 2000. Due to the smaller number of evaluations used during initialization in the bi-objective algorithms, the number of iterations in these methods is set to $2000 + \lceil \frac{d+d_{max}}{ipsize} \rceil$, where d is the dimensionality of the data set. Consequently, the number of evaluations used by all algorithms is approximately equal.

C. Data sets

The synthetic data sets used in our analysis have been previously described in [12]. They 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 (including k -means) [11]. The generator is applied to produce a number of small data sets with $k \in \{2, 4, 10\}$ and $d_1 \in \{2, 10\}$ (individual cluster sizes are uniformly distributed within the interval $\{10, \dots, 50\}$). We produce ten instances of each type. In our experiments, all ten data sets that are of dimensionality d_1 and contain k clusters are then grouped and referred to as the group of data d_1d-kc . Finally, a number of Gaussian noise variables $d_2 \in \{0, 100, 1000\}$ are added to all types of data, resulting in a total dimensionality $d = d_1 + d_2$. Hence, in total we obtain 18 different groups of data sets, which consist of 10 individual instances each.

In addition, we use real data sets taken from the Machine Learning Repository [7]. 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.

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 unlabelled and labelled data, which correspond to training and testing data respectively. Consistent with the principles of transductive inference [21], both the unlabelled and the labelled data are used during the classification process. For the experiments presented in the following, a fixed number of 5 labelled items are used per cluster. The remaining data items in every cluster are treated as unlabelled data.

D. Pre-processing

As suggested in [12], variables are normalized to a mean of zero and a standard deviation of one for all of the algorithms discussed in this paper.

E. Performance evaluation

In order to evaluate the quality of all solutions in the Pareto front, the Adjusted Rand Index is calculated for the unlabelled (testing) data. The use of the unlabelled data only ensures that the results obtained by the unsupervised, semi-supervised and supervised algorithms can be fairly compared. We then analyze the quality of the best solution identified by the different algorithms, that is we use external knowledge in order to select the best solution present in the Pareto front.

Of course, the results obtained in this way can be seen as overly optimistic, as external knowledge has been used to select the best solution from the Pareto front. In practice, the selection of good solutions may be more involved and one may need to compromise with solutions of lower quality. For our Pareto-based approach (method (1) above), we therefore additionally analyze the quality of our method of internal solution selection (described in Section III-E), that is we evaluate the quality of the solution that is most consistent with the prior knowledge, i.e. the one with the best score on objective (ii). In the case of draws, a single solution is chosen uniformly at random.

For each data set, a Wilcoxon Signed Rank test is applied to each pair of algorithms' results. This is a nonparametric test for differences between two paired (or matched) samples, as described in [5]. 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. 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. For the Machine Learning Repository results a Mann-Witney U test is used because the data are not paired.

V. RESULTS

Table II to Table V show the results obtained by the Pareto-based approach (using internal and external solution

selection) and by the four contestant methods (using external solution selection) on the synthetic and real data sets. Overall, these results tend to support the use of a multiobjective approach to semi-supervision, and we will now discuss individual aspects in isolation.

When considering the results obtained for the original data sets without noise (Table II), it is evident that the semi-supervised approaches outperform those based on unsupervised or supervised classification. However, the differences between the multiobjective approach and the MOEAs based on a linear and non-linear combination are not very pronounced. Specifically, all three algorithms generate the same solutions for those data sets containing two clusters, and it is only on the four and ten cluster data sets that a slight advantage of the multiobjective approach can be observed. This result can be explained by the implementation of our algorithms. For solution generation, all five algorithms rely on the use of the k -means algorithm. This means that the solution space 'seen' by the evolutionary algorithm is restricted to the space of possible k -means solutions, which corresponds to the set of local optima in terms of intra-cluster variance. For easy clustering problems containing a small number of clusters, only few such solutions exist (possibly only one), and no significant performance differences can therefore be observed between the different algorithms. The advantages of the multiobjective approach only become effective for complex data sets giving rise to a multitude of local optima.

The results in Table III and Table IV show the scalability of the different algorithms towards a large number of noise features. The results produced by the unsupervised and the supervised algorithm suffer significantly with increasing dimensionality, which is caused by random correlations arising from the large number of noise features. Interestingly, the performances of both the linear and the non-linear approaches also suffer, and these are sometimes outperformed by the unsupervised approach. In contrast to this, the performance of the multiobjective approach remains nearly unchanged, indicating that the explicit use of both types of information helps to overcome the effects of random correlations.

A comparison between the results of the Pareto-based approach using internal and external solution selection indicate a robust performance of the internal approach. While the best possible solution is not always selected, only a small performance deficit can be observed and, on most of the data sets, the results remain superior to those obtained by the other competing methods, which all use external solution selection. Note that the same scheme for solution selection is far less successful when used for the output of the semi-supervised algorithms based on a linear and non-linear combination (data not shown). The reason for this is that, for these methods, we observe a high number of ties when computing the Adjusted Rand Index values across the labelled data for all solutions in the Pareto front; and this leads to a uniformly random selection of solutions. We are not sure of the cause

TABLE II

RESULTS FOR SYNTHETIC DATA WITH 5 LABELLED ITEMS PER CLUSTER AND NO NOISE VARIABLES, AS DESCRIBED BY THE ADJUSTED RAND INDEX AND THE CARDINALITY OF THE CORRESPONDING FEATURE SET (AVERAGES OVER 21×10 RUNS). THE STATISTICALLY BEST PERFORMERS ARE IDENTIFIED IN BOLD FONT. SEE SECTION IV-E FOR INFORMATION ON THE STATISTICAL TESTING PROCEDURE.

Data set	Semisupervised								Unsupervised		Supervised	
	Pareto internal		Pareto external		Non-linear		Linear		rand	card	rand	card
	rand	card	rand	card	rand	card	rand	card				
2d-2c	0.976595	1.4	0.976595	1.4	0.976595	1.4	0.976595	1.4	0.964814	1.3	0.892309	1.1
2d-4c	0.801471	1.6	0.804371	1.7	0.77233	1.7	0.774316	1.7	0.720042	1.9	0.763406	1.61905
2d-10c	0.804914	2	0.805657	2	0.792276	2	0.789842	2	0.678451	2	0.783032	2
10d-2c	0.969954	3.0381	1	3.57143	1	1.98095	1	2.01429	0.987921	2.40476	0.756198	1.2
10d-4c	0.984351	8.37143	0.986177	8.61429	0.987721	8.70476	0.987929	8.9381	0.953076	8.28571	0.751055	2.64286
10d-10c	0.925253	9.31905	0.92609	9.35238	0.926912	9.52381	0.921481	9.54286	0.886644	9.42381	0.805114	7.01905

TABLE III

RESULTS FOR SYNTHETIC DATA WITH 5 LABELLED ITEMS PER CLUSTER AND 100 NOISE VARIABLES, AS DESCRIBED BY THE ADJUSTED RAND INDEX AND THE CARDINALITY OF THE CORRESPONDING FEATURE SET (AVERAGES OVER 21×10 RUNS). THE STATISTICALLY BEST PERFORMERS ARE IDENTIFIED IN BOLD FONT. SEE SECTION IV-E FOR INFORMATION ON THE STATISTICAL TESTING PROCEDURE.

Data set	Semi-supervised								Unsupervised		Supervised	
	Pareto internal		Pareto external		Non-linear		Linear		rand	card	rand	card
	rand	card	rand	card	rand	card	rand	card				
2d-2c	0.973279	1.90952	0.973279	1.90952	0.962298	2.03333	0.963907	2.33333	0.965729	1.99524	0.869866	1.06667
2d-4c	0.797522	1.62381	0.799454	1.74286	0.770584	1.87143	0.752478	2.10952	0.716954	2.09524	0.765561	1.68095
2d-10c	0.793927	2	0.794566	2	0.767603	2.0381	0.780997	2.00476	0.668204	2.01429	0.780976	2
10d-2c	0.976925	2.16667	1	2.36667	0.999354	2.31905	0.99774	2.44762	0.991038	3.34286	0.748006	1.00476
10d-4c	0.985242	10.0762	0.987452	10.3667	0.985946	10.3905	0.956643	14.9476	0.946657	9.72857	0.735556	2.65238
10d-10c	0.918145	10.719	0.919408	10.6619	0.908475	12.5952	0.826757	17.1143	0.887585	10.3	0.713125	5.59048

TABLE IV

RESULTS FOR SYNTHETIC DATA WITH 5 LABELLED ITEMS PER CLUSTER AND 1000 NOISE VARIABLES, AS DESCRIBED BY THE ADJUSTED RAND INDEX AND THE CARDINALITY OF THE CORRESPONDING FEATURE SET (AVERAGES OVER 21×10 RUNS). THE STATISTICALLY BEST PERFORMERS ARE IDENTIFIED IN BOLD FONT. SEE SECTION IV-E FOR INFORMATION ON THE STATISTICAL TESTING PROCEDURE.

Data set	Semi-supervised								Unsupervised		Supervised	
	Pareto internal		Pareto external		Non-linear		Linear		rand	card	rand	card
	rand	card	rand	card	rand	card	rand	card				
2d-2c	0.964599	1.31905	0.965952	1.34762	0.931931	1.95238	0.923802	1.8619	0.962801	1.40476	0.809899	1
2d-4c	0.796215	1.6381	0.79902	1.7381	0.734124	2.19048	0.695175	2.46667	0.704822	1.82381	0.664005	1.48095
2d-10c	0.795365	2	0.795924	2	0.743467	2.12857	0.782073	2.00476	0.663393	2.01905	0.776268	1.98571
10d-2c	0.977559	2.63333	1	2.84286	0.996654	2.8381	0.995715	2.87619	0.897159	3.24762	0.670767	1
10d-4c	0.973603	9.14762	0.976493	9.38095	0.972057	10.4952	0.918362	15.9429	0.849668	7.1619	0.609583	2.18571
10d-10c	0.914908	10.4524	0.916375	10.3524	0.902013	12.3476	0.797143	16.6762	0.770988	8.52857	0.610056	4.56667

TABLE V

RESULTS FOR REAL DATA WITH 5 LABELLED ITEMS PER CLUSTER, AS DESCRIBED BY THE ADJUSTED RAND INDEX AND THE CARDINALITY OF THE CORRESPONDING FEATURE SET (AVERAGES OVER 21 RUNS). THE STATISTICALLY BEST PERFORMERS ARE IDENTIFIED IN BOLD FONT. SEE SECTION IV-E FOR INFORMATION ON THE STATISTICAL TESTING PROCEDURE.

Data set	Semi-supervised								Unsupervised		Supervised	
	Pareto internal		Pareto external		Non-linear		Linear		rand	card	rand	card
	rand	card	rand	card	rand	card	rand	card				
iris	0.885697	1.66667	0.885697	1.66667	0.885697	1.7619	0.885697	1.71429	0.57576	1.14286	0.825934	1
wine	0.88453	7.33333	0.887708	7.14286	0.891989	7.28571	0.895155	7	0.461004	6	0.415449	2.61905
dermatology	0.858667	16.8571	0.858667	16.8571	0.859015	17.6667	0.851346	18.4286	0.607677	18.3333	0.576207	5.90476
zoo	0.924083	6.33333	0.929401	6.42857	0.859546	12.6667	0.865198	11.0952	0.866769	8.33333	0.915786	4.42857

of this effect but no straightforward, alternative method of internal solution selection seems possible to deal with this problem.

Our method of internal solution selection simply selects the solution that performs best under the supervised objective. Given this approach, one may wonder why the results obtained are not identical to those obtained by the supervised method, which optimizes exactly this supervised objective. The reason for this lies in the under-determination of the supervised problem in the presence of this small amount of training data, which leads to a multitude of solutions with an equivalent score under the supervised objective. The supervised method will assess these as equivalent and will therefore identify only one of these, which is likely to have poor generalization properties. In contrast, the Pareto front obtained by the semi-supervised method only contains the set of efficient trade-offs between the supervised and unsupervised objectives. This means that in the case of draws under the supervised objective, only the solution with the highest value under the unsupervised objective will be kept. The resulting solutions, which perform well both under the unsupervised and the supervised objective, are more likely to yield good generalization capabilities.

Note that the use of Pareto optimization in this way would be equivalent to the use of lexicographic ordering during the optimization process. However, the Pareto-based approach offers the future prospect of different methods of internal solution selection that can deal also with annotation errors (i.e. noisy class labels). In this case, when prior knowledge is not certain, we may wish to select solutions based on the shape of the Pareto front, an approach that has proven successful in our previous work on multiobjective clustering [11].

VI. CONCLUSION

In this paper, we have described a multiobjective evolutionary approach to semi-supervised feature selection. Experimental results on a large data test suite confirm results from the literature that indicate the advantages of a semi-supervised approach in scenarios where little prior knowledge is present. The results obtained also lend support to our hypothesis that a Pareto-based optimization of objectives related to both internal and external information may bear advantages over the optimization of a fixed linear or non-linear combination between the two.

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.

REFERENCES

[1] 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.

[2] M. Bilenko and R. J. Mooney. Adaptive duplicate detection using learnable string similarity measures. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 39–48. ACM Press, New York, NY, 2003.

[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] D. Cohn, R. Caruana, and A. McCallum. Semi-supervised clustering with user feedback. Technical Report Technical Report TR2003-1892, Cornell University, NY, 2003.

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

[6] 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.

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

[8] K. Deb and A. R. Reddy. Reliable classification of two-class cancer data using evolutionary algorithms. *BioSystems*, 72:111–129, 2003.

[9] 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.

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

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

[12] J. Handl and J. Knowles. Feature subset selection in unsupervised learning via multiobjective optimization. *International Journal on Computational Intelligence Research (to appear)*, 2006. Available from <http://dbk.ch.umist.ac.uk/handl/publications.html>.

[13] J. Handl and J. Knowles. On semi-supervised clustering via multiobjective optimization. Technical Report TR-COMPSYSBIO-2006-02, Manchester Interdisciplinary Biocentre, University of Manchester, UK, 2006. Available from <http://dbk.ch.umist.ac.uk/handl/publications.html>.

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

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

[16] 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.

[17] 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.

[18] J. M. Pena, J. A. Lozana, and P. Larranaga. An empirical comparison of four initialization methods for the k -means algorithm. *Pattern Recognition Letters*, 20(10):1027–1040, 1999.

[19] 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.

[20] 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.

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

[22] E. P. Xing, A. Y. Ng, M. I. Jordan, and S. Russell. Distance metric learning, with application to clustering with side-information. *Advances in Neural Information Processing Systems*, 15:505–512, 2003.

[23] K. Y. Yip, D. W. Cheung, and M. K. Ng. On discovery of extremely low-dimensional clusters using semi-supervised projected clustering. In *Proceedings of the 21st International Conference on Data Engineering*, pages 329–340. IEEE Press, Anaheim, CA, 2005.