# Dependency-based feature selection for clustering symbolic data

Luis Talavera

*Departament de Llenguatges i Sistemes Informàtics*
*Universitat Politècnica de Catalunya*
*Campus Nord, Mòdul C6, Jordi Girona 1-3*
*08034 Barcelona, Spain*
*talavera@lsi.upc.es*

**Abstract**

Feature selection is a central problem in data analysis that have received a significant amount of attention from several disciplines, such as machine learning or pattern recognition. However, most of the research has been addressed towards supervised tasks, paying little attention to unsupervised learning. In this paper, we introduce an unsupervised feature selection method for symbolic clustering tasks. Our method is based upon the assumption that, in the absence of class labels, we can deem as irrelevant those features that exhibit low dependencies with the rest of features. Experiments with several data sets demonstrate that the proposed approach is able to detect completely irrelevant features and that, additionally, it removes other features without significantly hurting the performance of the clustering algorithm.

*Key words:* Feature selection, clustering, data preprocessing.

## 1 Introduction

Many real world data analysis applications involve dealing with large and complex data sets containing both, many features and examples. Since, in this context, data have not been collected specifically for the data analysis task, often some form of preprocessing is required [9]. Among the various goals addressed by data preprocessing, *dimensionality reduction* or *feature selection* has been recognized as a central problem in data analysis [10], as reflected in the significant attention that this topic has recently received in the literature [3,4,14]. However, the vast majority of the research in feature selection has been carried out under the supervised learning paradigm, paying little attention to unsupervised learning problems. By contrast with supervised learning approaches, in unsupervised learning there are no target outputs associated with the inputs, and systems must resort to internal biases to decide which relationships should be represented in the output. Determining the relative importance of features in unsupervised environments is a difficult task given that the available knowledge is scarce.

Because of the absence of class labels in unsupervised data, the large body of feature selection methods proposed in supervised learning fails for this sort of data. Few attempts have been made in order to define unsupervised methods for feature selection, but two recent works are worth to mention. Devaney and Ram [6] proposed an attribute-incremental procedure for adding and removing features over an existing cluster hierarchy. Although there is little experimental evidence, their method appears to work fairly well. However, since they used the metric of a particular clustering system to evaluate the importance of features, it remains unclear if this method can be extended to work with other algorithms. Another promising proposal appears in [5], where an unsupervised entropy-based measure for ranking features is described. However, the empirical evaluation is carried out by comparing the selected features with the features selected by a supervised method and by using a supervised system. In order to assess the real capabilities of the method, further experiments involving the clustering task should be made.

In this paper, we introduce a simple method for ranking the importance of features in order to select a subset containing the most salient ones. Our method is based upon the assumption that, in the absence of class labels, we can deem as irrelevant those features that exhibit low dependencies with the rest of features. We describe a simple method using a feature dependence measure defined by Fisher [11]. Several experiments focused in the clustering task demonstrate that the method can readily detect irrelevant features without any kind of supervision.

## 2 Supervised and unsupervised feature selection

At a conceptual level, the feature selection problem is similar for both supervised and unsupervised learners. Considering feature selection as a heuristic search in a space of feature subsets, any method, supervised or unsupervised, requires an starting point in the space, a search strategy, an evaluation function and a stopping criterion [3]. Under this view, unsupervised feature selection methods could be designed by adapting existing supervised methods and adding a few task-specific modifications. However, in practice, the adaptation of the evaluation function is not straightforward, since all the existing criteria rely on assessing how well a given feature subset discriminates among a set of predefined classes that are not available for unsupervised learners. In fact, the problem stems from a more general issue related to the performance task associated with each type of learning. In supervised learning, the predictive accuracy over class labels is a widely accepted performance task, so it is relatively easy to design evaluation functions. On the contrary, there is a lack of a generally accepted performance task for unsupervised learners.

Let us consider the two main types of methods for feature selection in supervised learning, *filter* and *wrapper* models [14], in order to clarify the discussed problems. Filter models are independent of the induction algorithm that will use their output and they employ some metric dependent on intrinsic properties of the data. Typically, they measure the correlation of each feature with the class label by using distance, information or dependence measures [4]. Obviously, the absence of class labels makes infeasible to compute these sort of measures in unsupervised learning and, therefore, alternative measures not using class information need to be defined. Of course, most clustering systems are evaluated by using the resulting clustering in predicting the label of the objects in a test set, but, in this case, labels are used only for the external evaluation. The definition of a relevance metric turns out to be a complex problem since we need to decide what is going to be relevant for describing a set of classes that have not yet been created.

On the other hand, in the wrapper model, the feature selection algorithm works as a wrapper around the induction algorithm. Alternative feature subsets are evaluated by using the induction algorithm as a black box over the training data in order to obtain an estimate of future performance. Usually, performance is estimated by measuring the predictive accuracy over class labels. Therefore, similarly to filter methods, this method requires labels to be available during the training stage of learning. Again, an unsupervised learner can not have access to class labels and hence, can not perform the internal evaluation required by wrapper models.

We have seen that the main problem arises from the performance task tradi-

tionally used for assessing clustering systems and consisting in 'rediscovering' the underlying structure of the data. A different approach, outside the scope of this work, consists in considering a *flexible prediction* task that evaluates clustering systems regarding their ability to predict any unobserved feature [1,11,15], and not only a single target variable. By considering this later performance task, one could easily employ wrapper models, but the problem with filter models still remains.

## 3 A dependency-based measure for unsupervised feature selection

From previous discussion, we can conclude that wrapper models are not feasible for unsupervised feature selection when considering a class prediction task. Therefore, in order to implement a filter model, we have to decide some general characteristics of the training set allowing to select some features and discard others and implement these criteria in some measure. The central question is: what is relevant for a clustering task? Ideally, a formal definition of relevance would be needed in order to solve this question (see [3] for examples on definitions of relevance for supervised tasks). However, given the lack of research, in this paper we only aim to provide an informal and mainly empirical answer to this question, enabling further work towards a better formalization of the problem.

For the rest of the discussion, we will assume that a symbolic probabilistic clustering algorithm is used, which is not a very restrictive assumption considering the popularity of these sort of algorithms in machine learning [1,2,11]. To briefly introduce the notation used, we consider a set of symbolic (nominal) features $F = \{A_1, A_2, \ldots, A_n\}$, taking values $V_{ij}$ and data is represented as a set of vectors of feature-value pairs.

Clustering systems, either numerical or symbolic oriented, are intended to form partitions with high intra-cluster similarity and high inter-clustering dissimilarity. In other words, the goal is to form clusters having most feature values common to their members (*cohesion*) and few values common to members of other clusters (*distinctiveness*). In the context of symbolic probabilistic clustering, given a partition $(C_1, C_2, \ldots, C_k)$ and a certain value $V_{ij}$ for a feature $A_i$, cohesive clusters are those scoring a high $P(A_i = V_{ij} \mid C_k)$, while distinct clusters will score a high $P(C_k \mid A_i = V_{ij})$. Optimally, a good clustering should maximize those probabilities for a number of feature values. Fisher [12] argues that by rewarding clusters that simultaneously maximize both probabilities, at the same time, clusters formed around feature correlations are favored. Briefly and intuitively, if we form a cluster $C_k$ around a given value $V_{11}$, the cluster will have a high $P(A_1 = V_{11} \mid C_k)$ score because most of its members will exhibit this value. Conversely, since most of the other objects

have different values, we also obtain a high $P(C_k \mid A_1 = V_{11})$ score. Now, if $V_{11}$ is highly correlated with another value, say $V_{21}$, and the cluster captures this correlation, again, most members of $C_k$ will exhibit the value $V_{21}$. Hence, both, cohesion and distinctiveness provided by the value $V_{21}$ will be high and hence, the total amount of both measures for cluster $C_k$ will increase. Thus, cohesive and distinct categories tend to capture feature inter-correlations.

This result allows to conjecture that features that are not highly correlated with other features are not likely to play an important role in the clustering process, and can be deemed as irrelevant. Importantly, this conjecture is independent of any labeling of the data, so it can be readily applied to design a metric to rank the importance of the features without requiring such a labeling. Particularly, we propose a dependency measure defined by Fisher [11], although it was not originally proposed as a feature selection metric. The formulation of the measure is as follows. Let us consider the expected number of feature values that can be correctly predicted for a feature $A_M$ given knowledge of the value $V_{ij}$ for a feature $A_i$ as the conditional probability

$$\sum_j P(A_M = V_{j_M} \mid A_i = V_{ij})^2 \tag{1}$$

This expectation assumes a *probability matching* strategy [11] meaning that a feature value is guessed with probability $P(A_M = V_{j_M} \mid A_i = V_{ij})$ and that this guess is correct with the same probability.

Further, let us define the expected number of feature values than can be predicted for a feature $A_M$ with no knowledge , i.e., the base rate, as:

$$\sum_j P(A_M = V_{j_M})^2 \tag{2}$$

To assess how the knowledge of other features improves the prediction of a feature $A_M$, we can compute the gain that this knowledge provides by subtracting equation (2) from equation (1) and averaging the result for all the features and their values as follows:

$$\frac{\sum_i \sum_j P(A_i = V_{ij}) \sum_{j_M} [P(A_M = V_{Mj_M} \mid A_i = V_{ij})^2 - P(A_M = V_{Mj_M})]^2}{\mid \{i \mid A_i \neq A_M\} \mid} \tag{3}$$

where the leftmost factor gives higher weights to more frequent features. A simple filter model of feature selection can be implemented by calculating the feature dependence measure for each individual feature and then selecting the $k$ features with the highest value. This is a naive implementation from the point of view of searching the space of features, consisting in a single step

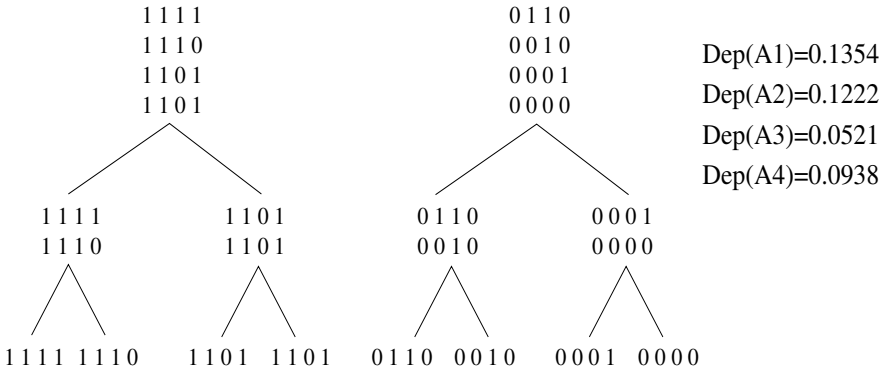


Fig. 1. An example relating cohesiveness, distinctiveness and dependency of features.

selection determined by $k$. However, we think that this simple ordering method can provide a good preliminary picture about the validity of our assumptions.

Figure 1 shows a simple example of the expected behavior of the presented measure. The figure shows the steps followed in constructing a two-class partition from a set of 8 objects. Each object is described by a vector of 4 binary features that will we called $A1 - A4$ according to their position in the vector. The clustering in the figure is built using a common agglomerative scheme that, at each step, merges the most similar pair of objects/clusters. These sort of methods tend to reward cohesive clusters, since they promote groupings of objects with similar values along all the features. At the same time, distinctiveness emerges as a result of forming cohesive clusters. For instance, both clusters in the top level of the hierarchy showed in Fig. 1, exhibit the same values for $A1$ along all the objects. The leftmost cluster exhibits the same pattern for $A2$, although in the other cluster this feature is somewhat less cohesive, since there is an object that has a different value than the rest of objects. $A4$ shows a similar lower cohesion but in both clusters. Finally, $A3$ exhibits a high variability inside of both clusters and, hence, it is not a cohesive feature at all. On the other hand, $A1$ is a distinctive feature, since it allows to perfectly differentiate between the two clusters. $A2$ is quite distinctive as well, although it fails for an individual object in the rightmost cluster. Again, $A4$ is partially useful as regards distinctiveness and $A3$ is completely irrelevant.

By applying the dependency measure to each feature, we obtain the scores showed next to the hierarchy in Fig. 1. Note that these scores allow to rank the features following the same criteria that we have obtained by manually examining the top partition. As mentioned before, this ranking can be obtained without any prior knowledge about labels or about how the objects are going to be clustered. This result demonstrates how a dependency measure can detect potentially cohesive and distinctive features. In this example, $A1$ and $A2$ are not only relevant from the standpoint of these two characteristics, but they also are more strongly correlated with other features than, for instance, $A3$.

While using information on feature dependences should help to identify irrelevant features fairly well, this method can exclude useful features in some cases. Suppose, for instance, a data set described by disjunctive rules of the form `if A1=V1j or A2=V2j then Ck`. Features $A1$ and $A2$ are clearly useful for describing the underlying classes in the data, but, in a disjunctive rule, if $A1$ takes a value that allows to decide class membership, $A2$ may take any value, so that no strong correlation between both features is guaranteed. Moreover, there can be a number of features moderately correlated with, lets say $A1$. In such a case, the proposed method could eventually remove $A2$, since it could score a lower correlation with other features. Note, however, that disjunctive descriptions are particularly hard to learn in an unsupervised manner and it is likely that, even without feature selection, a clustering system would find difficult to discover the original classes in these situations. We think that, basically, the method favors the original bias of the clustering system to form cohesive and distinct clusters. The presented situation reflects an example in which the original bias is not particularly adequate for a given learning task and, hence, a feature selection method promoting this bias may result innapropriate.

## 4 Experiments

In order to empirically asses the power of our feature selection scheme, we performed a set of experiments using several data sets from the UCI Repository and the well-known COBWEB clustering system [11]. Briefly, COBWEB is a hierarchical clustering system that constructs a probabilistic tree incrementally from a sequence of objects. Given an object and a current hierarchical clustering, the system categorizes the object by sorting it through the hierarchy from the root node down to the leaves. At each level, the learning algorithm evaluates the quality of the new clustering resulting from placing the object in each of the existing clusters, and the quality resulting from creating a new cluster covering the new object. In addition, the algorithm considers merging or splitting nodes to restructure the hierarchy in order to improve its quality.

We selected 8 data sets following different criteria. The first two are the LED and waveform data sets. Both are artificial datasets with added random features which turn out to be irrelevant for describing the underlying classes. Particularly, the LED data set has 7 original and 17 added features and the waveform data set has 21 original and 19 added features. These data allow to evaluate the behavior of our feature selection method when a relatively large number of features are irrelevant. The voting and mushroom data sets were selected because they are known to contain redundant features. Similarly, the credit and horse colic data sets have been used in previous work in supervised feature selection and may serve to obtain a preliminary compar-

Table 1
Data sets used in the experiments

| Data set | Objects | Features |
|---|---|---|
| LED-7+17 | 5000 | 24 |
| waveform-21+19 | 5000 | 40 |
| voting Records | 435 | 16 |
| mushroom | 8124 | 22 |
| crx | 690 | 15 |
| horse colic | 368 | 22 |
| audiology | 226 | 69 |
| breast cancer Wisconsin | 569 | 30 |

ative result. Finally, the audiology and breast cancer data sets were selected because they contain a high number of features (at least for the standards of the UCI Repository). Table 1 summarizes the main characteristics of each data set. Data sets containing numerical features have been discretized using the unsupervised method proposed by [16].

It is worth noticing at this point that the notion of irrelevance may be somewhat different for supervised and unsupervised learners. Although completely irrelevant features for supervised tasks such as those added in the LED and waveform data sets should be also irrelevant for clustering, it is likely that not all the features considered irrelevant in supervised environments should be considered as such for clustering tasks. In the absence of labels, clustering systems may need additional pieces of evidence to form categories, and, possibly, they will need larger feature subsets than supervised algorithms. Nevertheless, we should expect a reasonably reduction in dimensionality for data sets with a large number of features.

The results of the clustering process are evaluated by dividing the data set into a training and a testing set and running the algorithm on the training data with the class label masked out. Evaluation is performed by using the created clusters to predict the class label of the instances in the training set. Specifically, we conducted a 5x2cv paired t test as suggested in [7]. In this test, we perform 5 replications of a 2-fold cross-validation. In each replication the data are randomly divided into two equal-sized sets. The algorithm is then trained on each set and tested on the other.

Figure 2 shows the average error rate attained by using different $k$ values for our method and, therefore, different number of features. At a glance, it can be observed that our feature selection scheme does not significantly hurt performance, excepting when very low $k$ values are used. Particularly interesting

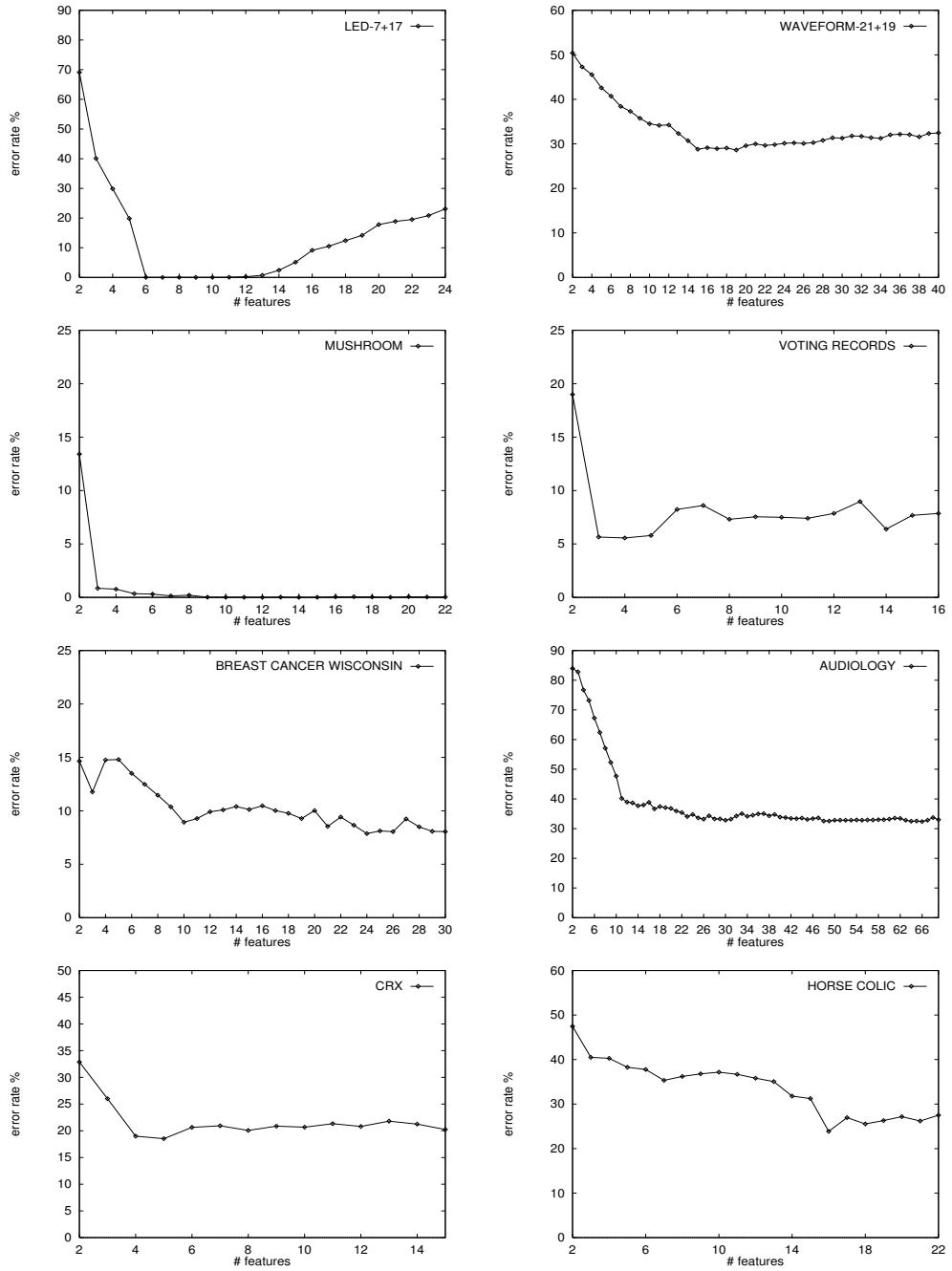


Fig. 2. Average error rates of 5x2 cross validation of COBWEB for different sizes of the feature set

are the LED and waveform results, where all the irrelevant features are correctly detected. Since the LED data set used in the experiment contains no noise, it is an extremely easy task for the clustering system to attain a 100% of predictive accuracy when the correct set of features is used. The waveform data set is much more difficult to learn for an unsupervised system, yet the best possible result can be obtained. The data sets with redundant features show also an interesting amount of reduction. Although it is known that in

Table 2
Selected results from several datasets and $k$ values. The $Conf$ column indicates the confidence on the hypothesis that the accuracy of COBWEB is better using all the features than using the indicated $k$ features

| Data set | Orig. feat. | $k$. | Orig. error | FS error | Conf. |
|---|---|---|---|---|---|
| Led-7+17 | 24 | 6 | 23.07 | 0.00 | 0.00 |
| waveform-21+19 | 40 | 19 | 32.46 | 28.62 | 0.00 |
| voting Records | 16 | 4 | 7.86 | 5.56 | 0.28 |
| mushroom | 22 | 9 | 0.03 | 0.02 | 0.20 |
| crx | 15 | 5 | 20.23 | 18.52 | 0.18 |
| horse colic | 22 | 16 | 27.50 | 23.86 | 0.11 |
| audiology | 69 | 30 | 33.01 | 32.83 | 0.10 |
| breast cancer W. | 30 | 10 | 8.05 | 8.93 | 0.34 |

both data sets, voting and mushroom, there is a very relevant feature that provides high accuracies [13], we cannot expect such an extreme result for unsupervised systems that do not have access to the class label and, hence, may need some additional features to be able to generate good clusters. The results on the remaining data sets show a similar behavior with the exception of the horse data set, in which accuracy degrades more quickly as more features are removed.

Table 2 shows a comparison between selected $k$ values for each data set and the results obtained using the full feature set. We selected the $k$ values yielding an improvement with reasonable statistical significance. As expected, in data sets with irrelevant features our method is able to remove all the irrelevant features and even some additional feature with a significant increase in accuracy. The results for the rest of the data sets show that around a 60% of features may be removed without a significant accuracy drop, with the mentioned exception of the horse data set. In some cases, using reduced feature sets results in significant higher accuracies at the 90% confidence level. Note that we have selected 'conservative' $k$ values indicating a high probability that feature selection improves accuracy. However, we think that the empirical results suggest that a more aggressive selection should not importantly hurt accuracy.

Although our experiments demonstrate that our method performs fairly well, we have not addressed the problem of how to choose an appropriate value for $k$, that is, how many features should be selected for each learning task. Figure 3 shows an example of the weights computed by the feature dependency measure for the LED and waveform data sets and provides some insight into this problem. Clearly, completely irrelevant features score very low, especially when compared with the other features and, accordingly, the histogram of the

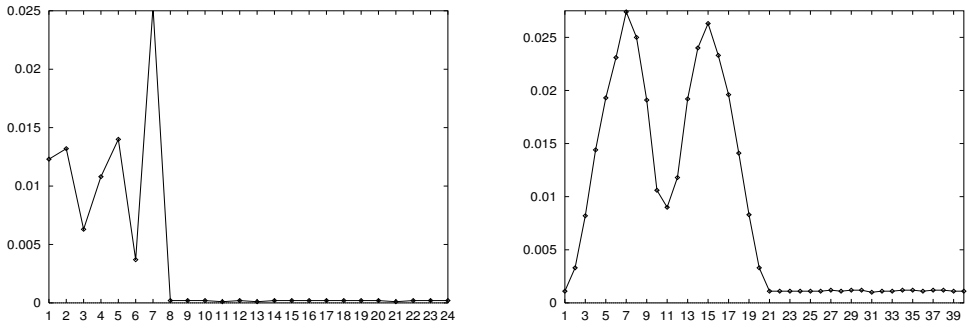


Fig. 3. Examples of feature dependences computed for the LED (left) and waveform (right) data sets

feature dependency scores appears to be a good source of information to decide how many features are worth to be removed. Moreover, we could think about automatic procedures such as clustering the points or using an unsupervised discretization method in order to generate a fixed number of feature subsets. These subsets would represent several degrees of relevance. From these sets we could use a conservative strategy removing only the features with very low relevances or a more aggressive approach selecting only the set of more important features.

## 5 Concluding remarks

We have presented an unsupervised feature selection method aimed to reduce the dimensionality of a symbolic data set prior to a clustering process. Our method is based upon the assumption that features that are little correlated with other features in the data are likely to be irrelevant. A straightforward implementation of the method has shown very powerful in experiments with several data sets, correctly identifying completely irrelevant features in the data sets in which these features are known. Experiments suggest that, in general, a reduction of the 60% of the original features may be possible without significantly hurting accuracy.

Despite the close relationship of the dependency measure used to rank the importance of features and the clustering system employed in the experiments, the method can be applied to any symbolic clustering algorithm. We have implemented the feature selection procedure as a preprocessing step and the method is decoupled from any particular clustering system. Furthermore, we have shown evidence that the measure can identify completely irrelevant features, so removal of these features should improve performance of different clustering algorithms. However, the removal of other features that are not clearly irrelevant could be more dependent on biases the particular algorithm. Future work should explore the effect of the presented feature selection meth-

ods in other clustering algorithms.

It is important to note that the feature selection strategy used is extremely simple. From the point of view of heuristic search, our method is a greedy search that only performs one step into the feature space to select the final feature set. Strategies such as stepwise sequential selection, which are very popular in the feature selection literature [8], may be combined with the dependency measure in order to see if more optimal feature subsets can be obtained. Furthermore, as a preprocessing step, the performance of the method for hierarchical clustering tasks may be limited by its global nature. A feature not scoring a very high –global– dependency, could become more relevant in a inner node of the cluster hierarchy when considering only a local region of the object space. Probably, these sort of features will not score very low dependences and they can be captured by selecting conservative $k$ values.

Finally, the dependency measure could be used to improve other performance tasks such as flexible prediction, that is, prediction of any missing feature and not only class labels. In fact, a version of the proposed metric has been previously used for feature selection and evaluated in flexible prediction tasks in [18], although not as a preprocessing step but embedded in the clustering system. Additionally we have employed this measure in such tasks elsewhere [17]. However, the question of what should be deemed as irrelevant in such a multiple inference task is still a more complex problem that the one we have addressed here and still remains open.

## References


[1] J. R. Anderson and M. Matessa. Explorations of an incremental, bayesian algorithm for categorization. *Machine Learning*, 9:275–308, 1992.

[2] G. Biswas, J. Weinberg, Q. Yang, and G. Koller. Conceptual clustering and exploratory data analysis. In L. Birnbaum and G. Collins, editors, *Proceedings of the Eight International Workshop on Machine Learning*, pages 591–595, Evanston, IL, 1991.

[3] A. L. Blum and P. Langley. Selection of relevant features and examples in machine learning. *Artificial Intelligence*, 97:245–271, 1997.

[4] M. Dash and H. Liu. Feature selection for classification. *Intelligent Data Analysis*, 1(3), 1997.

[5] M. Dash, H. Liu, and J. Yao. Dimensionality reduction for unsupervised data. In *Ninth IEEE International Conference on Tools with AI, ICTAI'97*, 1997.

[6] M. Devaney and A. Ram. Efficient feature selection in conceptual clustering. In *Proceedings of the Fourteenth International Conference on Machine Learning, ICML97*, Nashville, TN, 1997. Morgan Kaufmann.

[7] T. Dietterich. Statistical tests for comparing supervised classification learning systems. Technical report, Dept. of Computer Science, Oregon State University, 1996.

[8] J. Doak. An evaluation of feature-selection methods and their application to computer security. Technical Report CSE-92-18, Dept. of Computer Science, University of California at Davis, 1992.

[9] A. Famili, W. Shen, R. Webber, and E. Simoudis. Data preprocessing and intelligent data analysis. *Intelligent Data Analysis*, 1(1), 1997.

[10] U. M. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. From data mining to knowledge discovery: An overview. In U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, editors, *Advances in Knowledge Discovery and Data Mining*, pages 1–34. AAAI Press, Cambridge, Massachusetts, 1996.

[11] D. H. Fisher. Knowledge acquisition via incremental conceptual clustering. *Machine Learning*, 2:139–172, 1987.

[12] D. H. Fisher. *Knowledge acquisition via incremental conceptual clustering.* PhD thesis, University of California, Irvine, 1987.

[13] R. C. Holte. Very simple classification rules perform well on most commonly used datasets. *Machine Learning*, 11:63–91, 1993.

[14] R. Kohavi and G. H. John. Wrappers for feature subset selection. *Artificial Intelligence*, 97:273–324, 1997.

[15] P. Langley. *Elements of machine learning.* Morgan Kaufmann, San Francisco, CA, 1995.

[16] C. Li. Extending ITERATE conceptual clustering scheme in dealing with numeric data. Master's thesis, Vanderbilt University, 1995.

[17] L. Talavera. Feature selection as a preprocessing step for hierarchical clustering. In *Proceedings of the Sixteenth International Conference on Machine Learning, ICML99*, pages 389–397, Bled, Slovenia, 1999. Morgan Kaufmann.

[18] J. J. Furtado Vasco. Determining property relevance in concept formation by computing correlation between properties. In *Proceedings of the Tenth European Conference on Machine Learning, ECML98*, volume 1398 of *Lecture Notes in Artificial Intelligence*, pages 310–315, Chemnitz, Germany, 1998. Springer Verlag.