Feature Subset Selection and Order Identification for Unsupervised Learning

Jennifer G. Dy DY@ECN.PURDUE.EDU
Carla E. Brodley BRODLEY@ECN.PURDUE.EDU
School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47907 USA

Abstract

This paper explores the problem of feature subset selection for unsupervised learning within the wrapper framework. In particular, we examine feature subset selection wrapped around expectation-maximization (EM) clustering with order identification (identifying the number of clusters in the data). We investigate two different performance criteria for evaluating candidate feature subsets: scatter separability and maximum likelihood. When the "true" number of clusters k is unknown, our experiments on simulated Gaussian data and real data sets show that incorporating the search for k within the feature selection procedure obtains better "class" accuracy than fixing k to be the number of classes. There are two reasons: 1) the "true" number of Gaussian components is not necessarily equal to the number of classes and 2) clustering with different feature subsets can result in different numbers of "true" clusters. Our empirical evaluation shows that feature selection reduces the number of features and improves clustering performance with respect to the chosen performance criteria.

1. Introduction

For many feature selection problems, a human defines the features that are potentially useful and then a subset is chosen from the original pool of features using an automated feature selection algorithm. A significant body of research exists on methods for selecting features for supervised learning (Fukunaga, 1990; Kohavi & John, 1997), but until recently little attention has been paid to automatic feature selection for unsupervised learning. Feature selection is useful to limit redundancy of features, promote comprehensibility, and find clusters (or structures) hidden in high dimensional data. In unsupervised learning, our goal is to find the smallest feature subset that best uncovers "natural" groupings (clusters) from data according to some criterion. This is a difficult task because to find the feature subset that maximizes the performance criterion, we need the clusters to be defined. Moreover, unsupervised clustering needs the features or the variables that span the space we are trying to cluster. The problem is made more difficult when we do not know the number of clusters, k.

Our approach to feature selection for unsupervised learning is inspired by the wrapper approach for supervised learning (Kohavi & John, 1997), but rather than wrap the search for the best feature subset around a supervised induction algorithm, we wrap the search around a clustering algorithm. We introduce two methods: FSSEM (Feature Subset Selection wrapped around EM clustering) and FSSEM-k (FSSEM with order identification). In this paper, the term "EM clustering" represents the expectation-maximization (EM) algorithm (Dempster et al., 1977) applied to estimating the maximum likelihood parameters of a finite Gaussian mixture (McLachlan & Basford, 1988). Although we apply the wrapper approach to EM clustering, it can be applied to any clustering method.

2. Unsupervised Feature Selection Literature

To maintain the wrapper/filter model distinction used to characterize feature subset selection in supervised learning, we define the *wrapper* approach in unsupervised learning as applying the unsupervised learning algorithm to each feature subset in the search space and then evaluating the feature subset by a criterion function that utilizes the clustering result. *Filter* methods, on the other hand, use some intrinsic property of the data to select features without utilizing the clustering algorithm that will ultimately be applied.

A classic feature selection algorithm in the absence of class labels is to apply the Karhunen-Loéve transform (KLT) (Fukunaga, 1990). However, KLT is not a pure feature subset selection algorithm because it involves a transformation of the original feature space before selection.

Work in feature selection for unsupervised learning is relatively new. Most approaches are customized to a particular clustering algorithm. Devaney and Ram (1997) applied sequential forward and backward search. To evaluate each candidate subset, they measured the category utility of the clusters found by applying COBWEB (a hierarchical clustering algorithm) in conjunction with the feature subset. Talavera (1999) applied "blind" (similar to the filter) and "feedback" (analogous to the wrapper) approaches to COBWEB, and used a feature dependence measure to select features. Vaithyanathan and Dom (1999) formulated an objective function for choosing the feature subset and finding the optimal number of clusters for a document clustering problem using a Bayesian statistical estimation framework. Agrawal, et al. (1998) introduced a clustering algorithm (CLIQUE) which proceeds level-by-level starting from one feature upto the highest dimension or until no more feature subspaces with clusters (regions with high density points) are generated.

3. Feature Subset Selection and EM Clustering (FSSEM)

FSSEM wraps feature subset selection around the clustering algorithm. The basic idea is to search through feature subset space, evaluating each subset, F_t, by first clustering in space F_t using EM clustering and then evaluating the resulting clusters and feature subset using the chosen feature selection criterion. The two feature selection criteria investigated in this paper are discussed in Section 5. An exhaustive search of the 2^n possible feature subsets (n is the number of available features) for the subset that maximizes our selection criterion is computationally intractable. Therefore, a greedy search such as sequential forward or backward elimination (Fukunaga, 1990; Kohavi & John, 1997) is typically used. In the experiments reported, we applied sequential forward search. In the future, we plan to explore the effect of other search methods on FSSEM. Note that EM is initialized for each new feature subset.

In this paper, we assume that the data comes from a mixture model of multivariate Gaussians (McLachlan & Basford, 1988). We apply the EM algorithm to estimate the maximum likelihood mixture model parameters and the cluster probabilities of each data point. EM clustering results in "soft" clusters (i.e., each data point belongs to every cluster with some probability). Note that the framework introduced in this paper can easily be extended to other mixture probability distributions (McLachlan & Basford, 1988) and to other clustering methods, including graph theoretic approaches (Jain & Dubes, 1988).

The EM algorithm can become trapped at a local maximum, hence the initialization values are important. We used the sub-sampling initialization algorithm proposed by Fayyad, et al. (1998) with 10% sub-sampling and $J = 10$ sub-sampling iterations. After initializing the parameters, EM iterates until convergence (i.e., the likelihood does not change by 0.0001) or up to n (default 500) iterations whichever comes first. We limit the number of iterations because EM converges asymptotically, i.e., convergence is very slow near a maximum. EM estimation is constrained away from singular solutions in parameter space by limiting the diagonal elements of the component covariance matrices Σ_j to be greater than $\delta = 0.000001\sigma^2$ where σ^2 is the average of the variances of the unclustered data. Adding a scalar multiplied to the identity matrix (αI) to a positive semi-definite matrix where $\alpha > 0$ makes the final matrix positive definite (i.e., all eigenvalues are greater than zero and hence nonsingular).

4. Order Identification (FSSEM-k)

Unsupervised clustering is made more difficult when we do not know the number of clusters, k. A single perceptual class may be modeled better as a multiple Gaussian mixture than as a single Gaussian cluster. Furthermore, when we are also searching for the best subset of features, we run into a new problem: that the value of k *depends on the feature subset.* Figure 1 illustrates this point. In two dimensions (shown on the left) there are three clusters, whereas in one-dimension (shown on the right) there are only two clusters. The difficulty is in knowing which is better. Ultimately the only way to judge this is to use a criterion tied to the final use of the clustering.

FSSEM-k is just FSS wrapped around EM-k (EM clustering with order identification). For a given feature subset, we search for k and the clusters. EM-k currently applies the method by Bouman, et al. (1998), which adds a minimum description length (Rissanen, 1983) penalty term to the log-likelihood criterion. Our new objective function becomes: $F(k, \Phi) = \log(f(X|\Phi)) - \frac{1}{2}L\log(Nd)$ where N is the number of data points, d is the dimension, L is the number of real numbers needed to specify the parameters Φ and

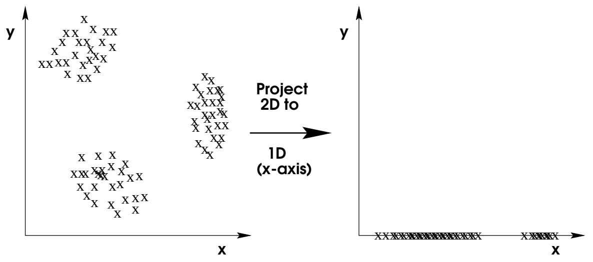

Figure 1. The number of cluster components varies with dimension.

$\log(f(X|\Phi))$ is the log-likelihood of our observed data X given the parameters Φ. Note that L and Φ vary with k. A penalty term is needed because the maximum likelihood estimate increases as more clusters are used. Without the penalty, the likelihood is at a maximum when each data point is considered as an individual cluster. We begin our search for k with a large number of clusters, K_{max}, and then sequentially decrement this number by one until only one cluster remains. We choose k to be the value that optimizes our criterion function. There are myriad ways to find the "optimal" number of clusters k with EM clustering (see Smyth (1996) for an overview).

5. Feature Selection Criteria

One of the factors that characterizes a feature selection algorithm is its performance (or feature evaluation) criterion. Here, we investigate two well-known measures: scatter separability and maximum likelihood. After describing each measure, we discuss the bias of each with respect to the dimension of the feature space. We conclude with an approach to ameliorate these biases via a dimension normalization procedure.

5.1 Scatter Separability Criterion

We investigate the scatter matrices and separability criteria used in discriminant analysis (Fukunaga, 1990) as our feature selection criterion. The criteria used in discriminant analysis assume that the features we are interested in are features that can separate the data into clusters that are unimodal and separated by their scatter means. Among the many possible separability criteria, we choose the $trace(S_w^{-1}S_b)$ criterion because it is invariant under any nonsingular linear transformation (Fukunaga, 1990). S_w is the within-class scatter matrix and S_b is the between class scatter matrix, and they are defined as follows:

$$\begin{aligned} S_w &= \textstyle\sum_{j=1}^{k} \pi_j E\{(X-\mu_j)(X-\mu_j)^T|\omega_j\} = \sum_{j=1}^{k} \pi_j \Sigma_j \\ S_b &= \textstyle\sum_{j=1}^{k} \pi_j (\mu_j - M_o)(\mu_j - M_o)^T \\ M_o &= E\{X\} = \textstyle\sum_{j=1}^{k} \pi_j \mu_j \end{aligned}$$

where π_j is the probability that an instance belongs to cluster ω_j, X is a d-dimensional random feature vector representing the data, k the number of clusters, μ_j is the sample mean vector of cluster ω_j, M_o is the total sample mean across all data points or instances in the data set, Σ_j is the sample covariance matrix of cluster ω_j, and $E\{\cdot\}$ is the expected value operator. S_w measures how scattered the samples are from their cluster means and the average covariance of each cluster. S_b measures how scattered the cluster means are from the total mean. We would like the distance between each pair of samples in a particular cluster to be as small as possible and the cluster means to be as far apart as possible with respect to the chosen similarity metric (Euclidean, in our case). $S_w^{-1}S_b$ is S_b normalized by the average cluster covariance. Hence, the larger the value of $trace(S_w^{-1}S_b)$ is, the larger the normalized distance between clusters is, which results in better cluster discrimination.

5.2 Maximum Likelihood (ML) Criterion

The scatter separability criterion does not utilize the additional information provided by EM clustering (the probability distribution estimate of the data). Separability is also based on a different underlying assumption from EM clustering. Separability is biased towards clusters with cluster means that are far apart, and biased against clusters with equal means, but different covariances. Separability is a general criterion that can be used for any clustering algorithm, but is more appropriate for distance-based clustering algorithms (like k-means). Maximum likelihood (ML), on the other hand, measures how likely our data are given the parameters and the model. Thus, it tells how well our model fits the data. Therefore, in addition to the criterion for clustering, we can employ ML to find the feature subset that models the data best using EM clustering. We choose the subset that maximizes this criterion.

5.3 Bias of Criterion Values to Dimension

Both criteria have biases with respect to the dimension of X. The separability criterion increases as the number of features (or dimension) increases, whereas ML decreases as the dimension increases. These biases occur even when the clustering assignments remain the same. The separability measure is biased this way because $trace(S_w^{-1}S_b)$ is basically adding d (the number of dimension) terms. Fukunaga (1990) proved that a criterion of the form $X_{d\times 1}^T S_{d\times d}^{-1} X_{d\times 1}$ monotonically increases with dimension, d, assuming the same clustering assignment. Dy (1999) relates $trace(S_w^{-1}S_b)$ to this proof. ML is biased the other way because the value

Table 1. Results for the synthetic data sets

	% CV Error			Clusters		
Method	2-Class	3-Class	4-Class	2-Class	3-Class	4-Class
FSSEM-TR	4.6 ±02.01	37.2 ±03.92	9.40 ±16.30	fixed at 2	fixed at 3	fixed at 4
FSSEM-k-TR	4.6 ±02.01	25.0 ±06.02	3.60 ±02.33	2.0 ±0.0	2.8 ±0.4	4.0 ±0.0
FSSEM-ML	47.0 ±14.54	39.8 ±20.15	19.60 ±31.25	fixed at 2	fixed at 3	fixed at 4
FSSEM-k-ML	55.6 ±03.88	54.6 ±17.78	79.40 ±06.14	1.0 ±0.0	1.4 ±0.8	1.0 ±0.0
EM	12.8 ±13.95	25.2 ±06.76	12.60 ±11.35	fixed at 2	fixed at 3	fixed at 4
EM-k	55.6 ±03.88	63.6 ±05.99	48.60 ±09.47	1.0 ±0.0	1.0 ±0.0	2.0 ±0.0

of the marginal density, $f(x)$, is always greater than or equal to $f(x,y)$ for any x. $f(x,y) = f(y|x)f(x)$, where $f(y|x)$ is the conditional density of y given x. Since $0 \leq f(y|x) \leq 1$, $f(x,y) = f(y|x)f(x) \leq f(x)$.

Given two feature subsets, F_1 and F_2, with different dimensions, clustering our data using subset F_1 leads to clustering C_1 and F_2 leads to clustering C_2. We now want to compare C_1 and C_2, but C_1 and C_2 exist in spaces of different dimension. To provide a fair comparison we propose the following heuristic normalization scheme.

Let $CRIT(F_i, C_j)$ be the criterion value using feature subset F_i to represent the data and C_j as the clustering assignment. $CRIT(\cdot)$ represents any arbitrary criterion function. For example, let $CRIT(\cdot)$ be the $trace(S_w^{-1}S_b)$ criterion. We compute S_w and S_b with X based on the feature subset F_i and μ's, σ's and π's based on the clustering assignments C_j and F_i. S_w and S_b are $d \times d$ matrices, where d is the number of features in F_i. We normalize the criterion value for C_1 as: $normalizedValue(C_1) = CRIT(F_1, C_1) \cdot CRIT(F_2, C_1)$, and the criterion value for C_2 as: $normalizedValue(C_2) = CRIT(F_2, C_2) \cdot CRIT(F_1, C_2)$. We would like to maximize our criterion values (ML or *trace*) as discussed in the previous subsections. Thus, if $normalizedValue(C_i) > normalizedValue(C_j)$, we choose clustering C_i and feature subset F_i. When the normalized criterion values are equal for C_i and C_j, we favor the clustering from the lower dimensional feature subset.

Normalization removes some of the bias of dimension because taking the product of $CRIT(F_i, C_i)$ and $CRIT(F_j, C_i)$ projects C_i to both dimensions. The normalized value, thus, focuses on the quality of the clusters obtained. When the clustering assignments using different feature subsets, F_1 and F_2, are the same (i.e., $C_1 = C_2$), the $normalizedValue(C_1)$ would be equal to the $normalizedValue(C_2)$ which is what we want. In the experiments reported in this paper, we applied the normalization criterion, because, a detailed analysis (Dy, 1999) showed that FSSEM without dimensionality bias correction results in more features than necessary for the trace criterion, and the selection of only one feature for the ML criterion.

6. Experimental Evaluation

Our experiments are designed to evaluate FSSEM and FSSEM-k along the dimensions of the ability to select relevant features, ability to correctly identify the order (k), and ability to find structure in the data. We first present experiments with synthetic data and then a detailed analysis of the FSSEM variants using two real-world data sets.

6.1 Synthetic Gaussian Mixture Data

Each of our synthetic data sets contains both "relevant" and "irrelevant" features, where relevant means that we created our k component mixture model using these features. Irrelevant features are generated as Gaussian normal random variables. The design of our first two synthetic data sets is similar to the simulated Gaussian structures reported in Smyth (1996). The first data set consists of two Gaussian clusters, both with covariance matrix, $\Sigma_1 = \Sigma_2 = I$ and means $\mu_1 = (0,0)$ and $\mu_2 = (0,3)$. There is considerable overlap between the two clusters and the added "noise" features increase the difficulty of the problem. For this data set, only feature 2 is considered relevant, since only feature 2 is needed to identify the two clusters. The second data set consists of three Gaussian clusters. Two clusters have means at $(0,0)$ but the covariance matrices are orthogonal to each other. The remaining cluster overlaps the tails on the right side of the other two clusters. The third data set has four clusters with means at $(0,0)$, $(1,4)$, $(5,5)$ and $(5,0)$ and covariances equal to I. For all three data sets, three Gaussian normal random "noise" features are added. We generated $N = 500$ data points and generated clusters that are of equal proportions. For these data sets, we set K_{max} to be five, five and ten respectively.

There are no standard measures for evaluating clusters in the clustering literature. Moreover, no single clustering assignment (or class label) explains every application (Jain & Dubes, 1988). Because we are working with synthetic data, we know the true mixture model and therefore the number of clusters, k, the true cluster assignments and the correct features. To measure performance, we use class error, which we define to be the number of instances misclassified divided by the total number of instances, assuming each instance within a cluster is classified to the majority class (determined by the training data). To compute class error, we assign each data point to its most likely cluster. When comparing clusterings with different number of clusters, one should not use training error. Class error based on training decreases with an increase in the number of clusters, k, with the trivial result of 0% error when each data is a cluster. To ameliorate this problem, we use cross-validation error (in particular, ten-fold cross-validation). Aside from class error, we also compute for the average number of clusters, the average number of features selected, feature precision and recall. Recall is the number of relevant features in the selected subset divided by the total number of relevant features. Precision is the number of relevant features in the selected subset divided by the total number of features selected.

Due to space limitations, we report only the ten-fold cross-validated class error (CV error) and the number of clusters found in Table 1. Complete results are found in Dy (1999). TR refers to the *trace* or scatter separability criterion, and ML refers to the maximum likelihood criterion.

Looking first at FSSEM-k-TR compared to FSSEM-TR, we see that including order identification (FSSEM-k-TR) with feature selection results in lower CV error for the trace criterion. For two data sets, FSSEM-k-TR had significantly lower CV error than FSSEM-TR. Adding the search for k within the feature subset selection search allows the algorithm to find the relevant features (an average of 0.97 feature recall for FSSEM-k-TR versus 0.85 for FSSEM-TR). This is because the best number of clusters depends on the chosen feature subset. For example, on closer examination, we noted that on the three-class problem that when k is fixed at three, the clusters formed by feature 1 are better separated than clusters that are formed by features 1 and 2 together. As a consequence, FSSEM-TR did not select feature 2. When k is made variable during the feature search, FSSEM-k-TR finds two clusters in feature 1. When feature 2 is considered with feature 1, three or more clusters are found resulting in higher separability.

EM had lower CV error than EM-k due to knowing the correct number of clusters. Both EM and EM-k had poorer performance than FSSEM-k-TR, because of the retained noisy features. FSSEM-k-TR performed better than FSSEM-k-ML in terms of CV error. The average feature recall and precision for all data sets for FSSEM-k-TR are 0.97 and 0.64 respectively. FSSEM-k-ML did not perform well with respect to CV error because it favors features that have unimodal distributions (like our noise features). FSSEM-TR and FSSEM-k-TR were biased toward separable clusters identified by our defined relevant features. FSSEM-ML and FSSEM-k-ML, on the other hand, were biased toward data with fewer clusters i.e., data that is more Gaussian in distribution (our defined noise features). Fixing k on the ML criterion improved CV error performance, because it increased the chances of arriving at a local minima on the noise features (more than one cluster, each with a different centroid) allowing the relevant features (with k equal to the true number of clusters) to be selected. Nevertheless, fixing k did not counter ML's bias towards noisy unimodal features (reflected by an average recall of 0.53).

6.2 Two Real-World Data Sets

To illustrate FSSEM on real data, we present results for two data sets: ionosphere (Blake & Merz, 1998) and a high resolution computed tomography images of the lungs (HRCT-lung) data set (Dy et al., 1999). See Dy (1999) for experiments on additional data sets. Although for each data set the class information is known, we remove the class labels during training.

Unlike synthetic data, we do not know the "true" number of (Gaussian) clusters for real-world data sets. Each class may be composed of many Gaussian clusters. Moreover, the clusters may not even have a Gaussian distribution. To see whether the clustering algorithms found clusters that correspond to classes (wherein a class can be multi-modal), we compute the class error in the same way as for the synthetic Gaussian data. On real data sets, we do not know the "relevant" features. Hence, we cannot compute precision and recall and therefore report only the average number of features selected and the average number of clusters found.

Table 2 reports the ten-fold cross-validation error and the number of clusters found by the different algorithms. For both data sets, we set K_{max} equal to ten. The Ionosphere data and HRCT-lung data have two and five labeled classes respectively. Note that even though we use class error as a measure of cluster performance, we should not let it misguide us in its interpretation. Cluster quality or interestingness is difficult to measure because it depends on the particular appli-

Table 2. Results for the Ionosphere and HRCT data sets.

	% CV Error		Clusters	
Method	Ionosphere	HRCT-lung	Ionosphere	HRCT-lung
FSSEM-TR	33.61 ±10.57	36.4 ±7.84	fixed at 2	fixed at 5
FSSEM-k-TR	26.22 ±07.35	34.0 ±6.07	7.2 ±1.2	6.8 ±1.0
FSSEM-ML	34.75 ±11.47	35.8 ±5.90	fixed at 2	fixed at 5
FSSEM-k-ML	20.82 ±08.12	36.0 ±6.13	8.0 ±0.4	6.7 ±0.6
EM	23.10 ±10.76	31.6 ±5.20	fixed at 2	fixed at 5
EM-k	24.21 ±08.17	37.2 ±5.53	1.9 ±0.3	1.0 ±0.0

cation. This is a major distinction between unsupervised clustering and supervised learning. Here, class error is just *one interpretation of the data.* We can also measure cluster performance in terms of the *trace* criterion and the ML criterion. Naturally, FSSEM-k-TR and FSSEM-TR were better than the rest in terms of *trace*; and, FSSEM-k-ML and FSSEM-ML were better than the rest in terms of maximum likelihood. Choosing either TR or ML depends on your application goals. If you are interested in finding the features that best separate the data, use FSSEM-k-TR. If you are interested in finding features that model Gaussian clusters best, use FSSEM-k-ML.

FSSEM-k-ML performed best in terms of CV error for the ionosphere data. Figure 2a presents a scatter plot of the ionosphere data on the two best features chosen by FSSEM-k-ML together with the means (in $\circ$'s) and covariances (in ellipses) discovered. Figure 2b shows the original data projected onto the same two features with the class labeled means and covariances. Figure 2c shows a scatter plot of the two best features chosen by FSSEM-k-TR and Figure 2d the corresponding scatter plot of the original data. Observe that *trace* favored the cluster in Figure 2c because the clusters are well separated. On the other hand, FSSEM-k-ML favored the clusters in Figure 2a which are more Gaussian and allow overlap. Since the ML clustering matches the ionosphere class labels more closely, FSSEM-k-ML performed better with respect to CV error. FSSEM-k-ML obtained better CV error than EM and EM-k. EM and EM-k used 32 features[1] whereas FSSEM-k-ML and FSSEM-k-TR used only 1.9 features on average. Fewer features made it possible for us to visualize the scatter plots in these dimensions. Order identification improved the performance for both criteria, because the true number of clusters appears to be five to six, even through there are only two classes in the ionosphere data set.

[1]Features 1 and 2 are discarded, because their values are either constant or discrete throughout the data. Constant features and discrete features with discrete levels less than or equal to the number of clusters produce infinite likelihood for a finite Gaussian mixture model.

For the HRCT lung data, FSSEM-k-TR performed better than FSSEM-k-ML in terms of CV error. Figure 3a presents a scatter plot of the HRCT-lung data on the two best features chosen by FSSEM-k-TR. Figure 3b shows the original data projected onto the same two features with the class labeled means and covariances. Observe that the clusters found by FSSEM-k-TR are well separated and match the class labels well. FSSEM-k-ML selected a single feature, which resulted in highly overlapping clusters. HRCT-lung is a difficult data set due to its skewed class distribution (approximately 62.8% of the data is from the disease Centrilobular Emphysema). Because of this, even though EM-k discovered only one cluster, its class error (which is equal to the error using a majority classification rule) is close to the values obtained by the other methods. The high dimensions obscure the HRCT-lung's classes and result in EM-k finding only one cluster. EM with k set to five performed better than FSSEM-k-TR in terms of CV error, but we do not always know the "true" number of clusters. In addition, EM uses 110 features, whereas FSSEM-k-TR uses an average of only 1.1 features.

When we examined our initial cluster result for HRCT-lung, we noticed the features selected by FSSEM-TR were the row and column centroid locations of the pathology region identified by a radiologist for each image. These features clustered the data in terms of location (regions on the upper left, lower left, upper right and lower right side of the image). Although these features result in well-separated clusters, they do not discriminate disease classes. So we subsequently removed them (resulting in our current data set of 110 features) and ran the algorithm again. Now, FSSEM-TR picked gray level histogram features and texture features which are relevant for discriminating the diseases. Although, we discarded the row and column features, we learned that our data is well separated in terms of location. Feature selection on unsupervised data is thus useful not only for discovering relevant features, but also for discovering unwanted features that produce systematic structures.

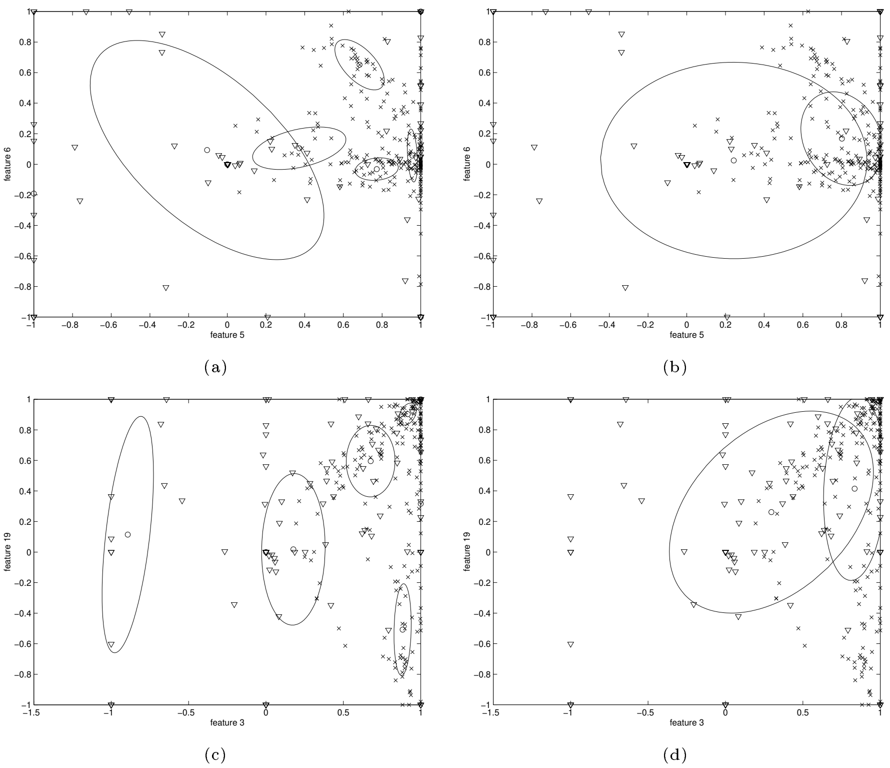

Figure 2. The scatter plots on ionosphere data using the two best features chosen by FSSEM-k-ML for (a) and (b), and the two best features chosen by FSSEM-k-TR for (c) and (d). + and $\bigtriangledown$ represent the different class assignments. $\circ$ are the cluster/class means, and the ellipses are the covariances. (a) and (c) are the clusters discovered by FSSEM-k-ML and FSSEM-k-TR respectively. (b) and (d) present the means and covariances using labeled classes.

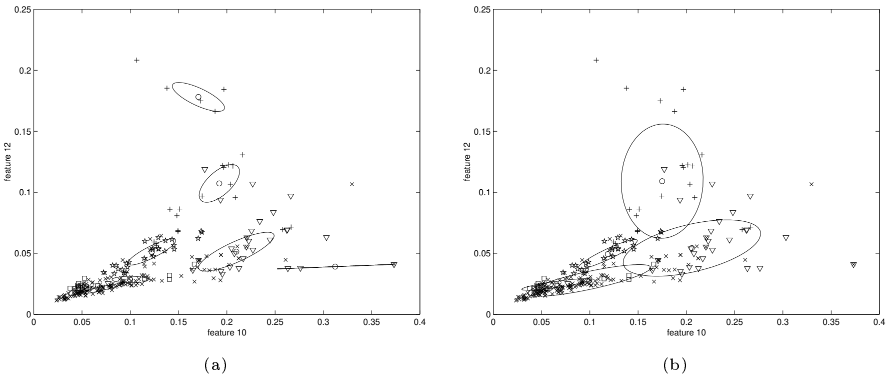

Figure 3. The scatter plots on HRCT-lung data using the two best features chosen by FSSEM-k-TR. +, $\bigtriangledown$, $\star$, $\times$ and the squares represent the different class assignments. $\circ$ are the cluster/class means, and the ellipses are the covariances. (a) shows the clusters found by FSSEM-k-TR. (b) shows the means and covariances using labeled classes.

7. Conclusion

In this paper, we introduced a wrapper framework for performing feature selection, clustering and order identification concurrently. We also compared two different feature selection criteria: scatter separability and maximum likelihood (ML). These criteria have different assumptions, biases and limitations. The separability based measure prefers feature subsets whose cluster centroids are far apart. ML prefers feature subsets that lead to clusters that fit the Gaussian model best. Cluster separation is not required to maximize the ML criterion, which can lead to overlapping clusters. In our experiments with synthetic data, the *trace* separability criterion performed better than the ML criterion. This result makes sense given that the data was designed to be well separated in the relevant feature subsets. Our results on the HRCT and Ionosphere data sets point out the utility in using feature subset selection to learn more about your data and the importance of picking a criterion tied to your clustering objectives. Finally, our results showed that incorporating order identification into the feature subset selection process led to better results than fixing k to be the true number of classes. There are two reasons: 1) the number of classes is not necessarily equal to the number of Gaussian clusters, and 2) different feature subsets have different number of clusters.

In the future, we would like to investigate different clustering algorithms and different search strategies in the wrapper framework. The normalization scheme introduced, although effective and general, is ad hoc. We would like to investigate other methods of normalizing the biases of the different criteria. Moreover, further investigation needs to be done on how k affects the feature search. It is also interesting to investigate on whether the feature selection criterion should be the same as the criterion used for clustering. Initial experiments on this are found in Dy (1999).

Acknowledgments

We thank Dr. Craig Codrington and the ML-lunch group for constructive comments and discussions. This research is supported by NSF Grant No. IRI9711535, and NIH Grant No. 1 R01 LM06543-01A1.

References

Agrawal, R., Gehrke, J., Gunopulos, D., & Raghavan, P. (1998). Automatic subspace clustering of high dimensional data for data mining applications. *Proceedings ACM SIGMOD International Conference on Management of Data* (pp. 94–105). Seattle, WA: ACM Press.

Blake, C., & Merz, C. (1998). UCI repository of machine learning databases. *http://www.ics.uci.edu/~mlearn/MLRepository.html*.

Bouman, C. A., Shapiro, M., Cook, G. W., Atkins, C. B., & Cheng, H. (1998). Cluster: An unsupervised algorithm for modeling gaussian mixtures. *http://dynamo.ecn.purdue.edu/~bouman/software/cluster*.

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal Royal Statistical Society, Series B*, *39*, 1–38.

Devaney, M., & Ram, A. (1997). Efficient feature selection in conceptual clustering. *Proceedings of the Fourteenth International Conference on Machine Learning* (pp. 92–97). Nashville, TN: Morgan Kaufmann.

Dy, J. G. (1999). *Preliminary Report: Feature Selection for Unsupervised Learning*. Unpublished manuscript, School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN.

Dy, J. G., Brodley, C. E., Kak, A., Shyu, C. R. S., & Broderick, L. S. (1999). The customized-queries approach to CBIR using EM. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 400–406). Fort Collins, CO: IEEE Computer Society Press.

Fayyad, U., Reina, C., & Bradley, P. S. (1998). Initialization of iterative refinement clustering algorithms. *Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining* (pp. 194–198). New York: AAAI Press.

Fukunaga, K. (1990). *Statistical pattern recognition (second edition)*. San Diego, CA: Academic Press.

Jain, A. K., & Dubes, R. C. (1988). *Algorithms for clustering data*. Englewood Cliffs, NJ: Prentice Hall.

Kohavi, R., & John, G. H. (1997). Wrappers for feature subset selection. *Artificial Intelligence*, *97*, 273–324.

McLachlan, G. J., & Basford, K. E. (1988). *Mixture models, inference and applications to clustering*. New York: Marcel Dekker.

Rissanen, J. (1983). A universal prior for integers and estimation by minimum description length. *Annals of Statistics*, *11*, 416–431.

Smyth, P. (1996). Clustering using Monte Carlo cross-validation. *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining* (pp. 126–133). Portland, OR: AAAI Press.

Talavera, L. (1999). Feature selection as a preprocessing step for hierarchical clustering. *Proceedings of the Sixteenth International Conference on Machine Learning* (pp. 389–397). Bled, Slovenia: Morgan Kaufmann.

Vaithyanathan, S., & Dom, B. (1999). Model selection in unsupervised learning with applications to document clustering. *Proceedings of the Sixteenth International Conference on Machine Learning* (pp. 433–443). Bled, Slovenia: Morgan Kaufmann.