Unsupervised Model Selection via Evolutionary Local Search

YongSeog Kim, W. Nick Street and Filippo Menczer
Management Sciences Department, Henry B. Tippie College of Business,
University of Iowa, Iowa City, IA 52242 USA
{yong-s-kim,nick-street,filippo-menczer}@uiowa.edu

Abstract

In this paper we consider the applicability of evolutionary multi-objective algorithms to the problem of unsupervised feature selection. Feature subset selection is important not only for the insight gained from determining relevant modeling variables but also for the improved understandability, scalability, and possibly, accuracy of the resulting models. We use ELSA, an evolutionary local selection algorithm that maintains a diverse population of solutions that approximate the Pareto front in a multi-dimensional objective space. Each evolved solution represents a feature subset and a number of clusters; a standard EM algorithm is applied to learn the parameters of the given number of clusters based on the selected features. Experimental results on both real and synthetic data show that the method can consistently identify a relevant subset of input features and an appropriate number of clusters. This results in models with better and clearer semantic relevance.

1 Introduction

Feature selection is the process of choosing a subset of the original predictive variables by eliminating redundant and uninformative ones. By extracting as much information as possible from a given data set while using the smallest number of features, we can save significant computing time and often build models that generalize better to unseen points. Further, it is often the case that finding a predictive subset of input variables is an important problem in its own right.

We adopt the wrapper model [Kohavi and John, 1997] of feature selection, which requires two components: a search algorithm that explores the combinatorial space of feature subsets, and one or more criterion functions that evaluate the quality of each subset based directly on the predictive model. Most feature selection research has focused on heuristic search approaches, such as sequential search [Kittler, 1986], nonlinear optimization [Bradley *et al.*, 1998], and genetic algorithms [Yang and Honavar, 1998]. These methods considered feature selection in a supervised learning context, evaluating potential solutions in terms of predictive accuracy. We instead wish to find natural grouping of the examples in the feature space via *clustering* or *unsupervised learning*. Clustering may be performed using methods such as K-means [Duda and Hart, 1973], expectation maximization (EM) [Dempster *et al.*, 1977], or optimization models [Bradley *et al.*, 1997]. Recently a set of novel clustering algorithms have been proposed in the database community [Zhang *et al.*, 1997; Guha *et al.*, 1998]. For instance, Agrawal *et al.,* [1998] present an order-independent clustering algorithm, CLIQUE, that forms clusters in large data sets. In this paper, we use the standard EM algorithm with each solution's selected subset of features. By using the EM algorithm, we can avoid the dependency of distance-based quality measurements on the dimensionality of the selected feature space, as observed in [Kim *et al.*, 2000].

We use evolutionary algorithms (EAs) to intelligently search the space of possible feature subsets (and the number of clusters, K). While a number of multi-objective extensions of evolutionary algorithms have been proposed in recent years [Deb and Horn, 2000], most of them, such as the Niched Pareto Genetic Algorithm [Horn, 1997], employ computationally expensive selection mechanisms to favor dominating solutions and to maintain diversity. Instead, we use a new evolutionary algorithm that maintains diversity over multiple objectives by employing a *local* selection scheme.

After reviewing the EM algorithm in Section 2, in Section 3 we discuss our approach, illustrating the evolutionary algorithm, and describing how ELSA is combined with EM. Section 4 presents some experiments with synthetic and real data sets, and discusses the interpretation of the ELSA output to select a subset of good features.

2 EM algorithm

The expectation maximization algorithm [Dempster *et al.*, 1977] is one of the most often used statistical modeling algorithms [Cheeseman and Stutz, 1996] and often significantly outperforms other clustering methods [Meila and Heckerman, 1998]. The EM algorithm assumes that the patterns are drawn from one of several distributions, and the goal is to identify the parameters of each distribution and their number. Starting with an initial estimate of the parameters, it iteratively recomputes the likelihood that each pattern is drawn from a particular density function, and then updates the parameter estimates.

Formally, let x_n, $n = 1, \cdots, N$, be a data point and x_{nj} be the value of the j-th feature of x_n. Let d be the dimension of the *selected* feature set, J, and K be the number of clusters. If we model each cluster with a d-dimensional Gaussian distribution, we can approximate the data distribution by fitting K density functions c_k, $k = 1, \cdots, K$, to the data set $\{x_n | n = 1, \cdots, N\}$. The probability density function evaluated at x_n is the sum of all densities:

$$P(x_n) = \sum_{k=1}^{K} p_k \cdot c_k(x_n|\theta_k) \tag{1}$$

where the *a priori* probability p_k is the fraction of the data points in cluster k and $\sum_{k=1}^{K} p_k = 1,\ p_k \geq 0$. The functions $c_k(x_n|\theta_k)$ are the density functions for patterns of the cluster k and θ_k are parameters such as mean and variance vector, μ_k and Σ_k. The membership probability of pattern x_n in cluster k is computed as follows:

$$p_k(x_n) = \frac{p_k \cdot c_k(x_n|\theta_k)}{\sum_{i=1}^{K} p_i \cdot c_i(x_n|\theta_i)}. \tag{2}$$

Now, the original problem of finding clusters is reduced to the problem of how to estimate the parameters $\Theta = \{\theta_1, \cdots, \theta_K\}$ of the probability density. With the independence assumption among attributes within a given cluster, we can represent each density function as a product of density functions over each selected attribute $j = 1, \cdots, d$:

$$c_k(x_n|\theta_k) = \prod_{j \in J} c_{kj}(x_{nj}|\theta_{kj}) \tag{3}$$

where θ_{kj} represents the parameters of the j-th feature of cluster k.

Finally, the maximum (log) likelihood (ML) method [Duda and Hart, 1973] is used to maximize the probability of data set given a particular mixture model as follows:

$$L(\Theta) = \sum_{n=1}^{N} \log\left(\sum_{k=1}^{K} p_k \cdot c_k(x_n|\mu_k, \Sigma_k)\right) \tag{4}$$

The EM algorithm begins with an initial estimation of Θ and iteratively updates it in such a way that the sequence of $L(\Theta)$ is non-decreasing. We outline the standard EM algorithm in Figure 1.

```
while |L(Θ^t) − L(Θ^{t+1})| > ε
  for each pattern x_n, n ∈ {1, ···, N}
    p_k^t(x_n) = p_k^t · c_k(x_n | μ_k^t, Σ_k^t) / Σ_{i=1}^K p_i^t · c_i(x_n | μ_i^t, Σ_i^t)
  endfor
  for each cluster k ∈ {1, ···, K}
    p_k^{t+1} = Σ_{n=1}^N p_k^t(x_n),  μ_k^{t+1} = Σ_{n=1}^N p_k^t(x_n)·x_n / Σ_{n=1}^N p_k^t(x_n)
    Σ_k^{t+1} = Σ_{n=1}^N p_k^t(x_n)(x_n − μ_k^{t+1})(x_n − μ_k^{t+1})^T / Σ_{n=1}^N p_k^t(x_n)
  endfor
  t = t + 1
endwhile
```

Figure 1: The summary of an EM algorithm where $\epsilon > 0$ is a stopping tolerance. p_k^t, μ_k^t, and Σ_k^t represents the mixture model parameters of cluster k at iteration t.

3 Feature selection algorithm

3.1 Heuristic metrics for clustering

In our study we use three fitness criteria, described below. One of the criteria is inspired by statistical metrics and two by Occam's razor. Each objective is normalized into the unit interval and maximized by the EA.

$\boldsymbol{F_{accuracy}}$**:** This objective is meant to favor cluster models with parameters whose corresponding likelihood of data is higher. With estimated distribution parameters, μ_k and Σ_k, $F_{accuracy}$ is computed as follows:

$$F_{accuracy} = \frac{1}{Z_A} \sum_{n=1}^{N} \log\left(\sum_{k=1}^{K} p_k \cdot c_k(x_n|\mu_k, \Sigma_k)\right)$$

where Z_A is an empirically derived, data-dependent normalization constant.

$\boldsymbol{F_{clusters}}$**:** Other things being equal, fewer clusters make the model more understandable and avoid possible overfitting. We implement this with the criterion

$$F_{clusters} = 1 - \frac{K - K_{min}}{K_{max} - K_{min}}$$

where K_{max} (K_{min}) is the maximum (minimum) number of clusters the user chooses to consider.

$\boldsymbol{F_{complexity}}$**:** This objective is aimed at minimizing the number of selected features for easier interpretability of solutions as well as better generalization:

$$F_{complexity} = 1 - \frac{d-1}{D-1},$$

where D is the dimensionality of the full data set.

3.2 Evolutionary local selection algorithm

ELSA springs from algorithms originally motivated by artificial life models of adaptive agents in ecological environments [Menczer and Belew, 1996]. In these models an agent's fitness results from individual interactions with the environment, which contains other agents as well as finite shared resources. A more extensive discussion of the algorithm and its application to Pareto optimization problems can be found elsewhere [Menczer *et al.*, 2000]. Figure 2 outlines the ELSA algorithm.

Each agent (candidate solution) in the population is first initialized with some random solution and an initial reservoir of *energy*. The representation of an agent consists of $D + K_{max} - 2$ bits. D bits correspond to the selected features (1 if a feature is selected, 0 otherwise). The remaining bits are a unary representation of the number of clusters.[1] This representation is motivated by the desire to preserve the

[1]The cases of zero or one cluster are meaningless, therefore we count the number of clusters as $K = \kappa + 2$ where κ is the number of ones and $2 \leq K \leq K_{max}$.

```
initialize p_max agents, each with energy θ/2
while there are alive agents in Pop^i and t < T
  for each energy source c ∈ {1,···,C}
    for each v ∈ {0,···,1}
      E^c_envt(v) ← 2v p_max E_cost/C
    endfor
  endfor
  for each agent a in Pop^i
    a' ← mutate(crossover(a, randommate))
    for each energy source c ∈ {1,···,C}
      v ← F_c(a')/P_c(F_c(a'))
      ΔE ← min(v, E^c_envt(v))
      E^c_envt(v) ← E^c_envt(v) − ΔE
      E_a ← E_a + ΔE
    endfor
    E_a ← E_a − E_cost
    if (E_a > α)
      insert a, a' into new population, Pop^{i+1}
      E_a' ← E_a/2
      E_a ← E_a − E_a'
    else if (E_a > 0)
      insert a into new population, Pop^{i+1}
    endif
    t = t + 1
  endfor
  i = i + 1
endwhile
```

Figure 2: ELSA pseudo-code. See text for details.

regularity of the number of clusters under the genetic operators. Mutation and crossover operators are used to explore the search space. A mutation operator randomly selects one bit of an agent and mutates it. Our crossover operator takes two agents, a parent a and a random mate, and scans through every bit of the two agents. If it locates a different bit, it flips a coin to determine the offspring's bit. In this process, the mate contributes only to construct the offspring's bit string, which inherits all the common features of the parents.

The environment corresponds to the set of possible values for each of the criteria being optimized.[2] We have an energy source for each criterion, divided into bins corresponding to its values. When the environment is replenished, each criterion is allocated an equal share of energy, apportioned in proportion to the fitness values in order to bias the population toward more promising areas in objective space. Note that the total replenishment energy that enters the system at each iteration is such that we can maintain a population size of p_{max} on average.

In each iteration of the algorithm, an agent explores a candidate solution similar to itself; it is rewarded with some energy from the environment and taxed with a constant cost. To compute the energy intake of an agent, for each objective function, the environment scales the agent's fitness value by the number of agents sharing the corresponding bin. Candidate solutions receive energy only inasmuch as the environment has sufficient resources; if these are depleted, no benefits are available until the environment is replenished. Thus an agent is rewarded with energy for its high objective values, but also has an interest in finding unpopulated niches in objective space, where more energy is available. In the selection part of the algorithm, an agent compares its current energy level with a constant reproduction threshold α. If its energy is higher than α, the agent reproduces: the agent and its offspring that was just evaluated become part of the new population, each with half of the parent's original energy. If the energy level is positive but lower than α, only the parent agent joins the new population.

In order to assign energy to a solution, ELSA must be informed of clustering quality. In the experiments described here, the clusters to be evaluated are constructed based on the selected features using the EM algorithm described in Section 2. Each time a new candidate solution is evaluated, the corresponding bit string is parsed to get a feature subset J and a number of clusters K. The EM algorithm is given the projection of the data set onto J, uses it to form K clusters, and returns the three fitness criteria $F_{accuracy}$, $F_{clusters}$, and $F_{complexity}$.

4 Evaluation

It is difficult to evaluate the quality of an unsupervised learning algorithm, and feature selection problems present the added difficulties that the clusters depend on the dimensionality of the selected features. In order to evaluate our approach, we construct a moderate-dimensional synthetic data set, in which the distributions of the points and the significant features are known, while the appropriate clusters in any given feature subspace are not known. We evaluate the evolved solutions by their ability to discover five pre-constructed clusters in a ten-dimensional subspace. We also use a real data set for which we have knowledge about the clusters and the relevant features. In this case, we can evaluate the solutions both by examining the selected features and by judging the semantics of the resulting clusters.

For further comparisons we have implemented a greedy heuristic algorithm known as the *plus 2-take away 1 sequential selection* algorithm [Kittler, 1986] using $F_{accuracy}$ as the optimization criterion. It begins by finding the single dimension along which the objective is optimized. At each successive step, the algorithm adds an additional feature that, when combined with the current set, forms the best clusters. It then checks to see if the least significant feature in the current set can be eliminated to form a new set with superior performance. This iteration is continued until all the features have been added. We ran the algorithm for each of the values of K considered by ELSA.

4.1 Experiments on a synthetic data set

We evaluate our approach on a synthetic data set which has $n = 500$ points and $D = 30$ features. It is constructed so that the first 10 features are significant, with 5 "true" clusters consistent across these features. The next 10 features are Gaussian noise, with points randomly and independently assigned to 2 normal clusters along each of these dimensions. The remaining 10 features are white noise. The standard deviation of the normal distributions is $\sigma \approx 0.06$ and the means are themselves drawn from uniform distributions in the unit interval, so that the clusters may overlap. We present some 2-dimensional projections of the synthetic data set in Figure 3.

Individuals are represented by 36 bits, 30 for the features and 6 for K ($K_{max} = 8$). There are 15 energy bins

[2] Here, $C = 3$ criteria; continuous objectives are discretized.

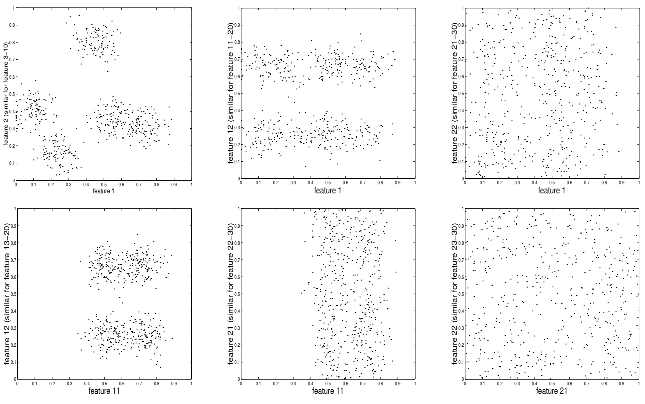

Figure 3: A few 2-dimensional projections of the synthetic data set.

for each of the energy sources, $F_{clusters}$, $F_{complexity}$, and $F_{accuracy}$. The values for the various ELSA parameters are: $\Pr(mutation)$ = 1.0, $\Pr(crossover)$ = 0.8, p_{max} = 100, E_{cost} = 0.2, α = 0.3, and T = 30,000.

For convenience, we call *Pareto front* the set of solutions with the highest value of $F_{accuracy}$ at every $F_{complexity}$ value for each different number of cluster K.[3] We found Pareto fronts based on all solutions evaluated and show them in Figure 4 for each K.

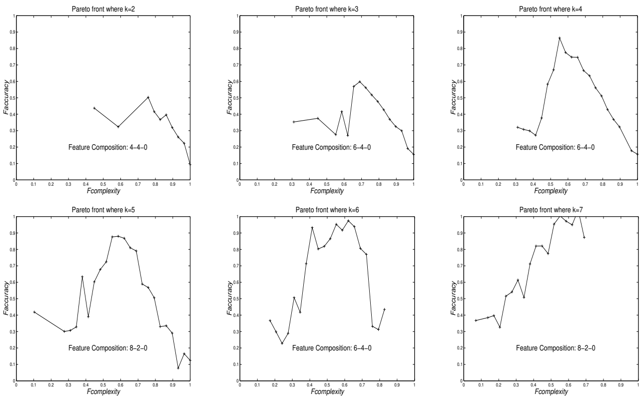

Figure 4: The Pareto fronts of ELSA/EM with the composition of features selected (see text).

We omit the Pareto front for K = 8 because of its inferiority in terms of clustering quality and incomplete coverage of search space. We expect the Pareto front for any reasonable K to take a typical shape: an ascent in the range of higher values of $F_{complexity}$ (lower complexity), and a descent for lower values of $F_{complexity}$ (higher complexity). This is reasonable because adding additional significant features will have a good effect on the the clustering quality with few preselected features. However, adding noisy features will have a negative effect on clustering quality. We note that the clustering quality and the coverage of search space improves as the used number of clusters approaches the "true" number of clusters, K = 5. ELSA/EM finds the correct number of clusters.

Note, however, that the selected features in the range of the first 10 features, $2/3 \leq F_{complexity} \leq 1$, are not necessarily all the "significant" features that we constructed. To quantify this notion we show in Figure 4 the composition of selected features, i.e., the number of significant-Gaussian noise-white noise features selected at $F_{complexity}$ = 0.69 where we could identify all the significant 10 features.[4] We attribute this finding to the fact that if one or more Gaussian noise features form good clusters with the pre-selected significant features, the clustering quality can be improved by adding these features. This is also consistent with the notion that not all strongly relevant features are selected and some weakly relevant features could be selected as "relevant" features [Kohavi and John, 1997]. The other Pareto fronts do not cover some ranges of the feature space because of either the agents' low $F_{clusters}$ for K = 7 or the agents' low $F_{accuracy}$ and $F_{complexity}$ for K = 2 and K = 3.

We also show snapshots of the Pareto front for K = 5 at intervals of every 3,000 solution evaluations in Figure 5.

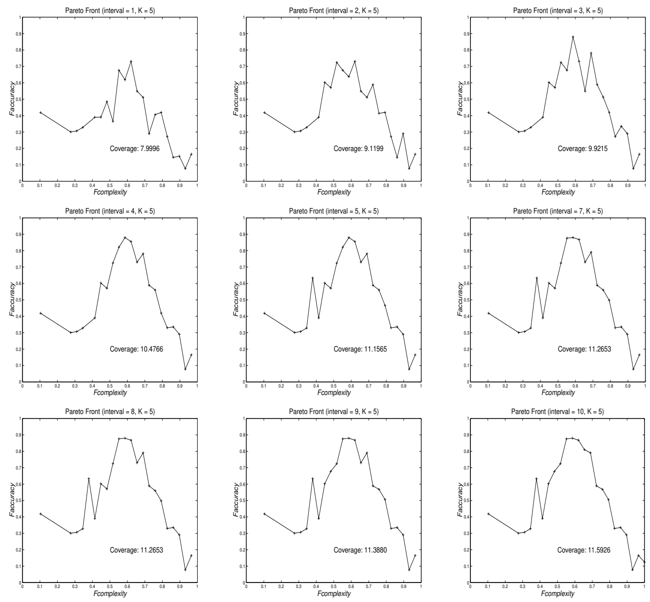

Figure 5: The trend of the Pareto front for K = 5 at intervals of every 3,000 solution evaluations. We omit the Pareto front for interval 6 because it is same as the Pareto front for interval 5.

Similarly to the ELSA/K-means model [Kim *et al.*, 2000], ELSA/EM explores a broader portion of the search space and thus identifies more accurate solutions across $F_{complexity}$ as more agents are evaluated. The *coverage* in the ELSA/EM

[3] Note that this is not the standard definition of Pareto front.

[4] For K = 2, we use $F_{complexity}$ = 0.76 when K = 2 which is the cloest value to 0.69 represented in Pareto front.

K		Number of selected features						
		2	3	4	5	6	7	Eval
2	ELSA/EM	52.6±0.3	56.6±0.6	92.8±5.2	100±0.0	100±0.0	100±0.0	5-0-1
	Greedy	51.8±1.3	52.8±0.8	55.4±1.1	56.6±0.4	62.8±3.2	80.2±8.5	
3	ELSA/EM	83.2±4.8	52±6.6	91.6±5.7	93.8±6.2	99±1.0	100±0.0	4-0-2
	Greedy	40.6±0.3	40.8±0.2	40.2±0.2	63.6±3.8	100±0.0	100±0.0	
4	ELSA/EM	46.2±2.2	–	50.6±0.6	89.6±5.9	52±1.0	60.6±5.1	4-2-0
	Greedy	27.8±0.8	27.8±0.4	29±0.4	29.6±0.9	38±4.4	74.2±3.5	
5	ELSA/EM	44.6±2.0	32.6±3.8	72±3.8	62.4±1.9	66.4±3.7	88 ±4.9	5-0-1
	Greedy	23±0.4	22.2±0.8	24.2±0.9	23.8±0.5	29.6±1.7	81.2±3.0	
Eval		3-0-1	3-1-0	4-0-0	4-0-0	3-0-1	1-1-2	18-2-4

Table 1: The average of classification accuracy (%) with standard error of five runs of ELSA/EM and greedy search. The "–" entry indicates that no solution founded by ELSA/EM and the last row and column shows the number of win-loss-tie cases of ELSA/EM compared with greedy.

model shown in Figure 5 is defined as the sum of $F_{accuracy}$ values over all $F_{complexity}$ values. We observe similar results for different numbers of clusters K.

We finally evaluated our approach in terms of classification accuracy and show our results in Table 4.1. We compute accuracy by assigning a class label to each cluster based on the majority class of the points contained in the cluster, and then computing correctness on *only those classes*, e.g., models with only two clusters are graded on their ability to find two classes. ELSA results represent individuals with less than eight features from Pareto fronts. ELSA consistently outperforms the greedy search on models with few features and few clusters, exactly the sort of models the algorithm was designed to find. For more complex models with more than 10 selected features (not shown), the greedy method is often better able to reconstruct the original classes. This is reasonable, since ELSA does not concentrate on this part of the search space.

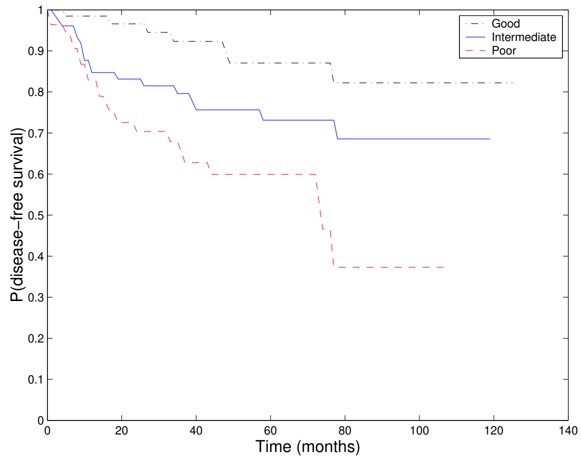

Figure 6: Estimated survival curves for the groups found by the ELSA-based clustering method on WPBC data.

4.2 Experimental results on WPBC data

In addition to the artificial data set discussed above, we also tested our algorithm on a real data set, the Wisconsin Prognostic Breast Cancer (WPBC) data [Mangasarian *et al.*, 1995]. This data set records 30 numeric features quantifying the nuclear grade of breast cancer patients at the University of Wisconsin Hospital, along with two traditional prognostic variables — tumor size and number of positive lymph nodes. This results in a total of 32 features for each of 198 cases.

For the experiment, individuals are represented by 38 bits, 32 for the features and 6 for K ($K_{max} = 8$). Other ELSA parameters are the same as those used in the previous experiment. We analyzed performance on this data set by looking for clinical relevance in the resulting clusters. Specifically, we can observe the actual outcome (time to recurrence, or known disease-free time) of the cases in the various clusters. Figure 6 shows a Kaplan-Meier estimate [Kaplan and Meier, 1958] of the true disease-free survival times for patients in the clusters represented in one evolved agent. A solution for this purpose was chosen with three clusters and the maximal curvature along the Pareto front.

The three groups displayed well-separated survival characteristics. Five-year recurrence rates were 13.49%, 25.73%, and 44.64% for the patients in the three groups. Further, the best prognostic group was statistically significantly different from the intermediate group ($p < 0.05$) and the intermediate group was well-differentiated from the poor group ($p < 0.03$). The chosen dimensions included a mix of nuclear morphometric features such as the mean radius and fractal dimension, the standard error of the radius, perimeter, area and smoothness, and the largest value of the radius. We note that one of the traditional medical prognostic factors, lymph node status, is not chosen. If this finding were supported by other experiments, the potentially hazardous surgical removal of lymph nodes from patients could be avoided.

5 Conclusions

In this paper we presented a novel evolutionary multi-objective local selection algorithm for unsupervised feature selection. We used ELSA to search for possible combination of features and numbers of clusters, with the guidance of the EM algorithm. The combination of a multi-objective search algorithm with unsupervised learning provides a promising framework for feature selection. We summarize our findings as follows.

- ELSA covers a large space of possible feature combinations while simultaneously optimizing the multiple criteria separately.
- The standard EM algorithm can be used to guide ELSA by evaluating the quality of a subset of features, while at the same time identifying the inherent numbers of clusters.
- Most importantly, in the proposed framework we can reliably select an appropriate clustering model, including significant features and the number of clusters. The result is a set of clusters that accurately models the data, and is more interpretable due to the reduced dimensionality.

From a data mining perspective, our algorithm can easily be used as a preprocessing step to determine an appropriate set of features (and number of clusters), allowing the application of iterative algorithms on much larger problems. In future work we would like to compare the performance of ELSA on the unsupervised feature selection task with other multi-objective EAs, using each in conjunction with the standard EM algorithm.

6 Acknowledgments

This work was supported in part by NSF grant IIS-99-96044.

References

[Agrawal *et al.*, 1998] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan. Automatic subspace clustering of high dimensional data for data mining applications. In *Proc. of the ACM SIGMOD Int'l Conf. on Management of Data*, pages 94–105, Seattle, WA, 1998.

[Bradley *et al.*, 1997] P. S. Bradley, O. L. Mangasarian, and W. N. Street. Clustering via concave minimization. In M. C. Mozer, M. I. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems 9*, pages 368–374, MA: Cambridge, 1997. MIT Press.

[Bradley *et al.*, 1998] P. S. Bradley, O. L. Mangasarian, and W. N. Street. Feature selection via mathematical programming. *INFORMS Journal on Computing*, 10(2):209–217, 1998.

[Cheeseman and Stutz, 1996] P. Cheeseman and J. Stutz. Bayesian classification system (AutoClass): Theory and results. In U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, editors, *Advances in Knowledge Discovery and Data Mining*, pages 153–180, San Francisco, CA, 1996. MIT Press.

[Deb and Horn, 2000] K. Deb and J. Horn. Special issue on multi-criterion optimization. *Evolutionary Computation Journal*, 8(2), 2000.

[Dempster *et al.*, 1977] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–38, 1977.

[Duda and Hart, 1973] R. Duda and P. Hart. *Pattern Classification and Scene Analysis*. John Wiley & Sons, New York, 1973.

[Guha *et al.*, 1998] S. Guha, R. Rastogi, and K. Shim. CURE: An efficient clustering algorithm for large databases. In *Proc. of ACM SIGMOD Int'l Conf. on Management of Data (SIGMOD98)*, pages 73–84, 1998.

[Horn, 1997] J. Horn. Multicriteria decision making and evolutionary computation. In *Handbook of Evolutionary Computation*. Institute of Physics Publishing, London, 1997.

[Kaplan and Meier, 1958] E. L. Kaplan and P. Meier. Nonparametric estimation from incomplete observations. *Journal of the American Statistical Association*, 53:457–481, 1958.

[Kim *et al.*, 2000] Y. Kim, W. N. Street, and F. Menczer. Feature selection in unsupervised learning via evolutionary search. In *Proc. of 6th ACM SIGKDD Int'l Conf. on Knowledge Discovery & Data Mining (KDD-00)*, pages 365–369, 2000.

[Kittler, 1986] J. Kittler. Feature selection and extraction. In Young Fu, editor, *Handbook of Pattern Recognition and Image Processing*, New York, 1986. Academic Press.

[Kohavi and John, 1997] R. Kohavi and G. H. John. Wrappers for feature subset selection. *Artificial Intelligence*, 97(1-2):273–324, 1997.

[Mangasarian *et al.*, 1995] O. L. Mangasarian, W. N. Street, and W. H. Wolberg. Breast cancer diagnosis and prognosis via linear programming. *Operations Research*, 43(4):570–577, July-August 1995.

[Meila and Heckerman, 1998] M. Meila and D. Heckerman. An experimental comparison of several clustering methods. Technical Report MSR-TR-98-06, Microsoft, Redmond, WA, 1998.

[Menczer and Belew, 1996] F. Menczer and R. K. Belew. From complex environments to complex behaviors. *Adaptive Behavior*, 4:317–363, 1996.

[Menczer *et al.*, 2000] F. Menczer, M. Degeratu, and W. N. Street. Efficient and scalable pareto optimization by evolutionary local selection algorithms. *Evolutionary Computation*, 8(2):223–247, Summer 2000.

[Yang and Honavar, 1998] J. Yang and V. Honavar. Feature subset selection using a genetic algorithm. *IEEE Intelligent Systems and their Applciations*, 13(2):44–49, 1998.

[Zhang *et al.*, 1997] T. Zhang, R. Ramakrishnan, and M. Livny. BIRCH: A new data clustering algorithm and its applications. *Data Mining and Knowledge Discovery*, 1(2):141–182, 1997.