Feature selection for high-dimensional genomic microarray data

Eric P. Xing[†] EPXING@CS.BERKELEY.EDU
Michael I. Jordan[†‡] JORDAN@CS.BERKELEY.EDU
Richard M. Karp[†] KARP@CS.BERKELEY.EDU
[†]Division of Computer Science, University of California, Berkeley, CA 94720
[‡]Department of Statistics, University of California, Berkeley, CA 94720

Abstract

We report on the successful application of feature selection methods to a classification problem in molecular biology involving only 72 data points in a 7130 dimensional space. Our approach is a hybrid of filter and wrapper approaches to feature selection. We make use of a sequence of simple filters, culminating in Koller and Sahami's (1996) Markov Blanket filter, to decide on particular feature subsets for each subset cardinality. We compare between the resulting subset cardinalities using cross validation. The paper also investigates regularization methods as an alternative to feature selection, showing that feature selection methods are preferable in this problem.

1. Introduction

Structural and functional data from analysis of the human genome have increased many fold in recent years, presenting enormous opportunities and challenges for machine learning. In particular, gene expression microarrays are a rapidly maturing technology that provide the opportunity to assay the expression levels of thousands or tens of thousands of genes in a single experiment (Shalon et al., 1996). These assays provide the input to a wide variety of statistical modeling efforts, including classification, clustering, and density estimation. For example, by measuring expression levels associated with two kinds of tissue, tumor or non-tumor, one obtains labeled data sets that can be used to build diagnostic classifiers. The number of replicates in these experiments are often severely limited, however; indeed, in the data that we analyze here (cf. Golub, et al., 1999), there are only 72 observations of the expression levels of each of 7130 genes. In this extreme of very few observations on very many features, it is natural—and perhaps essential—to investigate feature selection and regularization methods.

Feature selection methods have received much attention in the classification literature (Kohavi & John, 1997; Langley, 1994), where two kinds of methods have generally been studied—*filter* methods and *wrapper* methods. The essential difference between these approaches is that a wrapper method makes use of the algorithm that will be used to build the final classifier, while a filter method does not. Thus, given a classifier $\mathcal{C}$, and given a set of features F, a wrapper method searches in the space of subsets of F, using cross validation to compare the performance of the trained classifier $\mathcal{C}$ on each tested subset. A filter method, on the other hand, does not make use of $\mathcal{C}$, but rather attempts to find predictive subsets of the features by making use of simple statistics computed from the empirical distribution. An example is an algorithm that ranks features in terms of the mutual information between the features and the class label. Wrapper algorithms can perform better than filter algorithms, but they can require orders of magnitude more computation time. An additional problem with wrapper methods is that the repeated use of cross validation on a single data set can lead to uncontrolled growth in the probability of finding a feature subset that performs well on the validation data by chance alone. In essence, in hypothesis spaces that are extremely large, cross validation can overfit.

While theoretical attempts to calculate complexity measures in the feature selection setting generally lead to the pessimistic conclusion that exponentially many data points are needed to provide guarantees of choosing good feature subsets, Ng has recently described a generic feature selection methodology, referred to as FS-ORDERED, that leads to more optimistic conclusions (Ng, 1998). In Ng's approach, cross validation is used only to compare between feature subsets of different cardinality. Ng proves that this approach yields a generalization error that is upper-bounded by the logarithm of the number of irrelevant features.

In a problem with over 7000 features, filtering methods have the key advantage of significantly smaller computational complexity than wrapper methods, and for this reason these methods are the main focus of this paper. Earlier papers that have analyzed microarray data have also used filtering methods (Golub et al., 1999; Chow et al., in press; Dudoit et al., 2000). We show, however, that it is also possible to exploit prediction-error-oriented wrapper methods in the context of a large feature space. In particular, we adopt the spirit of Ng's FS-ORDERED approach and present a specific algorithmic instantiation of his general approach in which filtering methods are used to choose best subsets for a given cardinality. Thus we use simple filtering methods to carry out the major pruning of the hypothesis space, and use cross validation for final comparisons.

While feature selection methods search in the combinatorial space of feature subsets, regularization or shrinkage methods trim the hypothesis space by constraining the magnitudes of parameters (Bishop, 1995). Consider, for example, a linear regression problem in which the parameters θ_i are fit by least squares. Regularization adds a penalty term to the least squares cost function, typically either the squared L_2 norm or the L_1 norm. These terms are multiplied by a parameter λ, the *regularization parameter*. Choosing λ by cross validation, one obtains a fit in which the parameters θ_i are shrunk toward zero. This approach effectively restricts the hypothesis space, providing much of the protection against overfitting that feature selection methods aim to provide.

If the goal is to obtain small feature sets for computational or interpretational reasons, then feature selection is an obligatory step. If the goal is to obtain the best predictive classifier, however, then regularization methods may perform better than feature selection methods. Few papers in the machine learning literature have compared these approaches directly; we propose to do so in the current paper.

2. Feature selection

In this section we describe the feature selection methodology that we adopted. To summarize briefly, our approach proceeds in three phases. In the first phase we use *unconditional univariate mixture modeling* to provide an initial assessment of the viability of a filtering approach, and to provide a discretization for the second phase. In the second phase, we rank features according to an information gain measure, substantially reducing the number of features that are input to the third phase. Finally, in the third phase we use the more computationally intense procedure of *Markov blanket filtering* to choose candidate feature subsets that are then passed to a classification algorithm.

2.1 Unconditional mixture modeling

A useful empirical assumption about the activity of genes, and hence their expression, is that they generally assume two distinct biological states (either "on" or "off"). The combination of such binary patterns from multiple genes determines the sample phenotype. Given this assumption, we expect that the marginal probability of a given expression level can be modeled as a univariate mixture with two components (which includes the degenerate case of a single component). Representative samples of empirical marginals are shown in Figure 1.

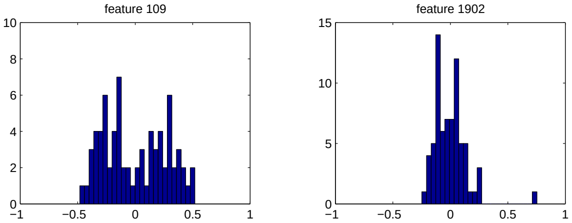


Figure 1. Two representative histograms of gene expression measurements. The x-axes represents the normalized expression level.

If the underlying binary state of the gene does not vary between the two classes, then the gene is not discriminative for the classification problem and should be discarded. This suggests a heuristic procedure in which we measure the separability of the mixture components as an assay of the discriminability of the feature.

Let $P(f_i \mid \theta_i)$ denote a two-component Gaussian mixture model for feature f_i, where θ_i denotes the means, standard deviations and mixing proportions of the mixture model. We fit these parameters using the EM algorithm. Note that each feature f_i is fit independently.

Suppose that we denote the underlying state of gene i as a latent variable $z_i \in \{0,1\}$. Suppose moreover that we define a decision $d(f_i)$ on feature f_i to be 0 if the posterior probability of $\{z_i = 0\}$ is greater than 0.5 under the mixture model, and let $d(f_i)$ equal 1 otherwise. We now define a *mixture overlap probability*:

$$\begin{aligned}\epsilon = P(z_i = 0)P(d(f_i) = 1|z_i = 0)\\ + \quad P(z_i = 1)P(d(f_i) = 0|z_i = 1).\end{aligned} \qquad (1)$$

If the mixture model were a true representation of the probability of gene expression, then the mixture overlap probability would represent the Bayes error of classification under this model. We use this probability as a heuristic surrogate for the discriminating potential of the gene, as assessed via its unconditional marginal.

Note that a mixture model can also be used as a quantizer, allowing us to discretize the measurements for a given feature. We simply replace the continuous measurement f_i with the associated binary value $d(f_i)$. This is in fact the main use that we make of the mixture models in the remainder of the paper. In particular, in the following section we use the quantized features to define an information gain measure.

2.2 Information gain ranking

We now turn to methods that make use of the class labels. The goal of these methods is to find a good approximation of the conditional distribution, $P(C \mid \mathbf{F})$, where $\mathbf{F}$ is the overall feature vector and C is the class label.

The *information gain* is commonly used as a surrogate for approximating a conditional distribution in the classification setting (Cover & Thomas, 1991). Let the class labels induce a *reference partition* $S_1, \ldots, S_C$. Let the probability of this partition be the empirical proportions: $P(T) = |T|/|S|$ for any subset T. Now suppose a test on feature F_i induces a partition of the training set into $E_1, \ldots, E_K$. Let $P(S_c|E_k) = P(S_c \cap E_k)/P(E_k)$. We define the information gain due to this feature with respect to the reference partition as:

$$\begin{aligned} I_{gain} &= H(P(S_1), \ldots, P(S_C)) \\ &- \sum_{k=1}^{K} P(E_k) H(P(S_1|E_k), \ldots, P(S_C|E_k)), \quad (2) \end{aligned}$$

where H is the entropy function. The information gain provides a simple initial filter with which to screen features. For example, one can rank all genes in the order of increasing information gain and select features conservatively via a statistical significance test (Ben-Dor et al., 2000).

To calculate the information gain, we need to quantize the values of the features. This is achieved in our approach via the unconditional mixture model quantization discussed in the previous section.

2.3 Markov blanket filtering

Features that pass the information gain filter are input to a more computationally intensive subset selection procedure known as *Markov blanket filtering*, a technique due to Koller and Sahami (1996)..

Let $\mathbf{G}$ be a subset of the overall feature set $\mathbf{F}$. Let $\mathbf{f}_G$ denote the projection of $\mathbf{f}$ onto the variables in $\mathbf{G}$. Markov blanket filtering aims to minimize the discrepancy between the conditional distributions $P(C|\mathbf{F} = \mathbf{f})$ and $P(C|\mathbf{G} = \mathbf{f}_G)$, as measured by a conditional entropy:

$$\Delta_{\mathbf{G}} = \sum_{\mathbf{f}} P(\mathbf{f}) D(P(C|\mathbf{F} = \mathbf{f}) \parallel P(C|\mathbf{G} = f_{\mathbf{G}}))$$

where $D(P\|Q) = \sum_x P(x) \log(P(x)/Q(x))$ is the Kullback-Leibler divergence. The goal is to find a small feature set $\mathbf{G}$ for which $\Delta_{\mathbf{G}}$ is small.

Intuitively, if a feature F_i is conditionally independent of the class label given some small subset of the other features, then we should be able to omit F_i without compromising the accuracy of class prediction. Koller and Sahami formalize this idea using the notion of a Markov blanket.

Definition 1 (Markov Blanket) *For a feature set* $\mathbf{G}$ *and class label* C*, the set* $\mathbf{M}_i \subseteq \mathbf{G}$ *(*$F_i \notin \mathbf{M}_i$*) is a* Markov Blanket *of* F_i *(*$F_i \in \mathbf{G}$*) if*

$$F_i \perp \mathbf{G} - \mathbf{M}_i - \{F_i\}, C \quad | \quad \mathbf{M}_i$$

The following proposition due to Koller and Sahami establishes the relevance of the Markov blanket concept to the measure $\Delta_{\mathbf{G}}$.

Proposition 2 *For a complete feature set* $\mathbf{F}$*, let* $\mathbf{G}$ *be a subset of* $\mathbf{F}$*, and* $\mathbf{G}' = \mathbf{G} - F_i$*. If* $\exists \mathbf{M}_i \subseteq \mathbf{G}$ *(where* $\mathbf{M}_i$ *is a Markov blanket of* F_i*), then* $\Delta_{\mathbf{G}'} = \Delta_{\mathbf{G}}$*.*

The proposition implies that once we find a Markov blanket of feature F_i in a feature set $\mathbf{G}$, we can safely remove F_i from $\mathbf{G}$ without increasing the divergence to the desired distribution. Koller and Sahami further prove that in a sequential filtering process in which unnecessary features are removed one by one, a feature tagged as unnecessary based on the existence of a Markov blanket $\mathbf{M}_i$ remains unnecessary in later stages when more features have been removed.

In most cases, however, few if any features will have a Markov blanket of limited size, and we must instead look for features that have an "approximate Markov blanket." For this purpose we define

$$\begin{aligned} \Delta(F_i|\mathbf{M}) = \sum_{f_{\mathbf{M}}, f_i} & P(\mathbf{M} = f_{\mathbf{M}}, F_i = f_i) \\ & D(P(C|\mathbf{M} = f_M, F_i = f_i) \, \| P(C|\mathbf{M} = f_M)) \quad (3) \end{aligned}$$

If $\mathbf{M}$ is a Markov blanket for F_i then $\Delta(F_i|\mathbf{M}) = 0$. Since an exact zero is unlikely to occur, we relax the condition and seek a set $\mathbf{M}$ such that $\Delta(F_i|\mathbf{M})$ is small. Note that if $\mathbf{M}$ is really a Markov blanket of F_i, then we have $P(C|\mathbf{M}, F_i) = P(C|\mathbf{M})$. This suggests an easy heuristic way to to search for a feature with an approximate Markov blanket.

Since the goal is to find a small non-redundant feature subset, and those features that form an approximate Markov blanket of feature D_i are most likely to be more strongly correlated to F_i, we construct a candidate Markov blanket for F_i by collecting the k features that have the highest correlations (defined by the Pearson correlations between the original non-quantized feature vectors) with F_i, where k is a small integer. We have the following algorithm as proposed in (Koller & Sahami, 1996):

```
Initialize
  - $\mathbf{G} = \mathbf{F}$
Iterate
  - For each feature $F_i \in \mathbf{G}$, let $\mathbf{M}_i$ be the set of
    $k$ features $F_j \in \mathbf{G} - \{F_i\}$ for which the correlations between $F_i$ and $F_j$ are the highest.
  - Compute $\Delta(F_i|\mathbf{M}_i)$ for each $i$
  - Choose the $i$ that minimizes $\Delta(F_i|\mathbf{M}_i)$, and
    define $\mathbf{G} = \mathbf{G} - \{F_i\}$
```

This heuristic sequential method is far more efficient than methods that conduct an extensive combinatorial search over subsets of the feature set. The heuristic method only requires computation of quantities of the form $P(C|\mathbf{M} = \mathbf{f}_M, F_i = \mathbf{f}_i)$ and $P(C|\mathbf{M} = \mathbf{f}_M)$, which can be easily computed using the discretization discussed in Section 2.1.

3. Classification algorithms

We used a Gaussian classifier, a logistic regression classifier and a nearest neighbor classifier in our study. In this section we provide a brief description of these classifiers.

3.1 Gaussian classifier

A Gaussian classifier is a generative classification model. The model consists of a prior probability π_c for each class c, as well as a Gaussian class-conditional density $\mathcal{N}(\mu_c, \Sigma_c)$ for class c.[1] Maximum likelihood estimates of the parameters are readily obtained.

Restricting ourselves to binary classification, the posterior probability associated with a Gaussian classifier is the logistic function of a quadratic function of the feature vector, which we denote here by x:

$$P(y = 1|x, \theta) = \frac{1}{1 + \exp\{\frac{1}{2}x^T \Lambda x - \beta^T x - \gamma\}}$$

where Λ, β and γ are functions of the underlying covariances, means and class priors. If the classes have equal covariance then Λ is equal to zero and the quadratic function reduces to a linear function.

3.2 Logistic regression

Logistic regression is the discriminative counterpart of the Gaussian classifier. Here we assume that the posterior probability is the logistic of a linear function of the feature vector:

$$P(y = 1|x, \theta) = \frac{1}{1 + e^{-\theta^T x}},$$

where θ is the parameter to be estimated. Geometrically, this classifier corresponds to a smooth ramp-like function increasing from zero to one around a decision hyperplane in the feature space.

Maximum likelihood estimates of the parameter vector θ can be found via iterative optimization algorithms. Given our high-dimensional setting, and given the small number of data points, we found that stochastic gradient ascent provided an effective optimization procedure. The stochastic gradient algorithm takes the following simple form:

$$\theta^{(t+1)} = \theta^{(t)} + \rho(y_n - \mu_n^{(t)})x_n,$$

where $\theta^{(t)}$ is the parameter vector at the tth iteration, where $\mu_n^{(t)} \equiv 1/(1 + e^{-\theta^{(t)T} x_n})$, and where ρ is a step size (chosen empirically in our experiments).

3.3 K nearest neighbor classification

We also used a simple K nearest neighbor classification algorithm, setting K equal to three. The distance metric that we used was the Pearson correlation coefficient.

4. Regularization methods

Regularization methods provide a popular strategy to cope with overfitting problems (Bishop, 1995). Let $l(\theta \mid \mathcal{D})$ represent the log likelihood associated with a probabilistic model for a data set $\mathcal{D}$. Rather than simply maximizing the log likelihood, we consider a "penalized likelihood," and define the following "regularized estimate" of the parameters:

$$\hat{\theta} = \arg\max_{\theta} \{l(\theta \mid \mathcal{D}) - \lambda||\theta||\},$$

where $||\theta||$ is an appropriate norm, typically the L1 or the L2 norm, and where λ is a free parameter known as the "regularization parameter." The basic idea is that the penalty term often leads to a significant decrease in the variance of the estimate, at the expense of a slight bias, yielding an overall decrease in risk. One can also take a Bayesian point of

[1] Note that when the covariance matrix Σ_c is diagonal, then the features are independent given the class and we obtain a continuous-valued analog of the popular naive Bayes classifier.

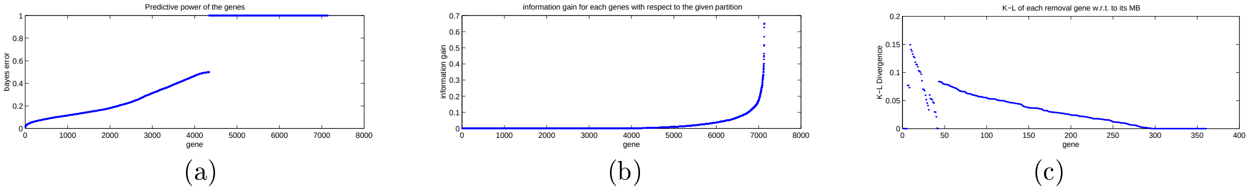


Figure 2. Feature selection using using a 3-stage procedure. (a) Genes ranked by ϵ (Eq. 1); (b) Genes ranked by I_{gain} (Eq. 2); (c) Genes ranked by $\Delta(F_i|\mathbf{M})$ (Eq. 3).

view and interpret the penalty term as a log prior, in which case regularization can be viewed as a maximum a posteriori estimation method.

The regularization parameter λ is generally set via some form of cross validation.

An L2 penalty is a rotation-invariant penalty, and shrinks the parameters along a ray toward the origin. Using an L1 penalty, on the other hand, shrinks the parameters toward the L1 ball, which is not rotation invariant. Some of the parameters shrink more quickly than others, and indeed parameters can be set to zero in the L1 case (Tibshirani, 1995). Thus an L1 penalty has something of the flavor of a feature selection method.

Parameter estimation is straightforward in the regularization setting, with the penalty term simply contributing an additive term to the gradient. For example, in the case of logistic regression with an L2 penalty, we obtain the following stochastic gradient:

$$\theta^{(t+1)} = \theta^{(t)} + \rho \left((y_n - \mu_n^{(t)}) x_n - \lambda \theta^{(t)} \right),$$

where the shrinkage toward the origin is apparent. In the case of Gaussian classifier, the 'regularized' ML estimate of θ can be easily solved in closed form.

5. Experiments and Results

In this section, we report the results of analysis of the data from a microarray classification problem. Our data is a collection of 72 samples from leukemia patients, with each sample giving the expression levels of 7130 genes (Golub et al., 1999). According to pathological/histological criteria, these samples include 47 type I Leukemias (called ALL) and 25 type II Leukemias (called AML). The samples are split into two sets by the provider, with 38 (ALL/AML=27/11) serving as a training set and the remaining 34 (20/14) as a test set. The goal is to learn a binary classifier (for the two cancer subtypes) based on the gene expression patterns.

5.1 Filtering results

Figure 2(a) shows the mixture overlap probability ϵ (defined by Eq. 1) for each single gene in ascending order. It can be seen that only a small percentage of the genes have an overlap probability significantly smaller than $\epsilon \ll 0.5$, where 0.5 would constitute random guessing under a Gaussian model if the underlying mixture components were construed as class labels.

In Figure 2(b) we present the information gain that can be provided by each individual gene with respect to the reference partition (the Leukemia class labels), compared to the partition obtained from the mixture models. Only a very small fraction of the genes induce a significant information gain. We take the top 360 genes from this list and proceed with (approximate) Markov blanket filtering.

Figure 2(c) displays the values of $\Delta(F_i|\mathbf{M}_i)$ (cf. Eq. 3) for each F_i, an assessment of the extent to which the approximate Markov blanket $\mathbf{M}_i$ subsumes information carried by F_i and thus renders F_i redundant. Genes are ordered in their removal sequence from right to left. Note that the redundancy measure $\Delta(F_i|\mathbf{M}_i)$ increases until there are fewer than 40 genes remaining. At this point $\Delta(F_i|\mathbf{M}_i)$ decreases, presumably because of a compositional change in the approximate Markov blankets of these genes compared to the original contents before many genes were removed. The increasing trend of $\Delta(F_i|\mathbf{M}_i)$ then resumes.

The fact that in a real biological regulatory network the fan-in and fan-out will generally be small provides some justification for enforcing small Markov blankets. In any case, we have to keep the Markov blankets small to avoid fragmenting our small data set.

5.2 Classification results

Figure 3 shows training set and test set errors for each of the three different classifiers. For each feature subset cardinality (the abscissa in these graphs), we chose a feature subset using Markov blanket filtering. This is a classifier-independent method, thus the feature subsets are the same in all three figures.

The figures show that for all classifiers, after an initial coevolving trend of the training and testing curves for low-dimensional feature spaces (the dimensionality differs for the different classifiers), the

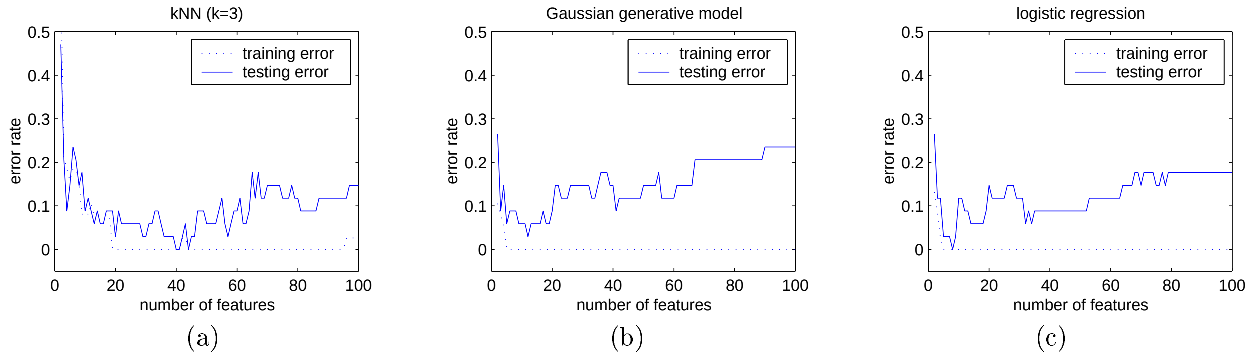


Figure 3. Classification in a sequence of different feature spaces with increasing dimensionality due to inclusion of gradually less qualified features. (a) Classification using kNN classifier; (b) Classification using a quadratic Bayesian classifier given by a Gaussian generative model; (c) A linear classifier obtained from logistic regression. All three classifiers use the same 2-100 genes selected by the three stages of feature selection.

classifiers quickly overfit the training data. For the logistic linear classifier and kNN, the test error tops out at approximately 20 percent when the entire feature set of 7130 genes is used. The generative Gaussian quadratic classifier overfits less severely in the full feature space. For all three classifiers, the best performance is achieved only in a significantly lower dimensional feature space. Of the three classifiers, kNN requires the most features to achieve its best performance.

Figure 3 shows that by an optimal choice of the number of features it is possible to achieve error rates of 2.9%, 0%, and 0% for the Gaussian classifier, the logistic regression classifier, and kNN, respectively. Of course, in actual diagnostic practice we do not have the test set available, so these numbers are optimistic. To choose the number of features in an automatic way, we make use of leave-one-out cross validation on the training data. That is, for each cardinality of feature subset, given the feature subset chosen by our filtering method, we choose among cardinalities by cross validation. Thus we have in essence a hybrid of a filter method and a wrapper method—the filter method is used to choose feature subsets, and the wrapper method is used to compare between best subsets for different cardinalities.

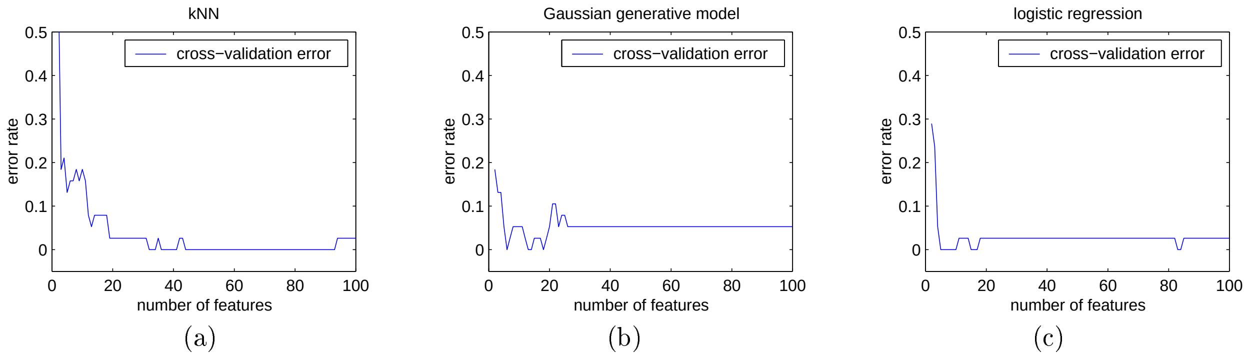
The results of leave-one-out cross validation are shown in Figure 4. Note that we have several minima for each of the cross-validation curves. Breaking ties by choosing the minima having the smallest cardinality, and running the resulting classifier on the test set, we obtain error rates of 8.8%, 0%, and 5.9% for the Gaussian classifier, the logistic regression classifier, and kNN, respectively.

We also compared prediction performance when using the unconditional mixture modeling (MM) filter alone and the information gain (IG) filter alone (in the latter case, using the discretization provided by the first phase of mixture modeling). The results for the logistic regression classifier are shown in Figure 5. As can be seen, the number of features determined by cross-validation using the MM filter is 20 (compared to 8 using the full Markov blanket filtering) and the resulting classifier also has a higher test set error (5.9% versus 0%). For the IG filter, the selected number of features is 59, and the test set error rate is significantly higher (13.5%). The latter result in particular suggests that it is not sufficient to simply performance a "relevance check" to select features, but rather that a redundancy reduction method such as the Markov blanket filter appears to be required. Note also that using the MM filter alone results in better performance than using the IG filter alone. While neither approach performs as well as Markov Blanket filtering, the MM filter has the advantage that it does not require class labels. This opens up the possibility of doing feature selection on this data set in the context of unsupervised clustering (see Xing & Karp, 2001).

Table 1. Performance of classification based on randomly selected features (200 trials)

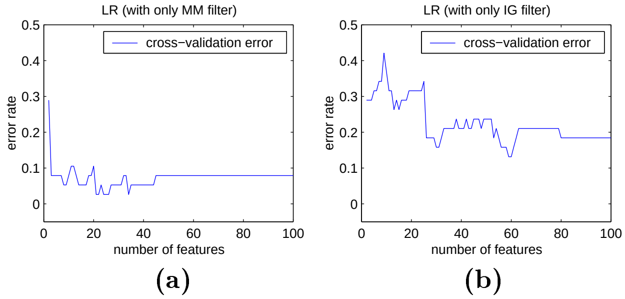
	training error (%)			test error (%)		
classifier	Max	Min	Average	Max	Min	Average
kNN	50.0	7.9	27.1	50.0	8.8	35.6
Gaussian	28.9	5.3	14.2	64.7	14.7	35.6
Logistic	31.6	2.6	17.4	50.0	20.6	35.6

In some high-dimensional problems, it may be possible to bypass feature selection algorithms and obtain reasonable classification performance by choosing random subsets of features. That this is not the case in the Leukemia data set is shown by the results (Table 1). In the experiments reported in this table, we chose ten randomly selected features for each classifier. The performance is poorer than in the case of explicit feature selection.



(a) (b) (c)

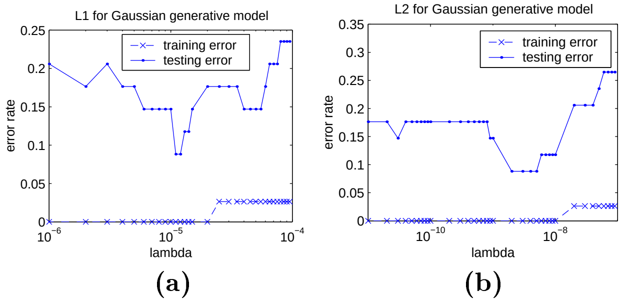
Figure 4. Plots of leave-one-out cross validation error for the three classifiers.



(a) **(b)**

Figure 5. Plots of leave-one-out cross validation error for the logistic regression classifier with (a) only the unconditional mixture modeling filter and (b) only the information gain filter.

5.3 Regularization versus feature selection



(a) **(b)**

Figure 6. Training set and test set error as a function of the regularization parameter λ. The results are for the Gaussian classifier using the (a) L1 and (b) L2 penalties.

Figure 6 shows the results for the regularized Gaussian classifier using the L1 and L2 penalties. Similar results were found for the logistic regression classifier.

By choosing an optimal value of λ based (optimistically) on the test set, we obtain test set errors of 8.8% and 8.8% for the Gaussian classifier for the L1 and L2 norm respectively. For the logistic regression classifier, we obtain test set errors of 17.6% and 20.1%.

These errors are higher than those obtained with explicit feature selection. Indeed, a comparison of Figures 3 and 6, which show the range of test set performance achievable from the feature selection and the regularization approaches, respectively, show that the feature selection curves are generally associated with smaller error. Given that the regularization approach can, in the worst case, leave us with all 7130 features, we feel that feature selection provides the better alternative for our problem.

6. Discussion and Conclusion

We have shown that feature selection methods can be applied successfully to a classification problem involving only 38 training data points in a 7130 dimensional space. This problem exemplifies a situation that will be increasingly common in applications of machine learning to molecular biology. Microarray technology makes it possible to put the probes for the genes of an entire genome onto a chip, such that each data point provided by an experimenter lies in the high-dimensional space defined by the size of the genome under investigation.

In high-dimensional problems such as these, feature selection methods are essential if the investigator is to make sense of his or her data, particularly if the goal of the study is to identify genes whose expression patterns have meaningful biological relationships to the classification problem. Computational reasons can also impose important constraints. Finally, as demonstrated in smaller problems in the extant literature on feature selection (Kohavi & John, 1997; Langley, 1994), and as we have seen in the high-dimensional problem studied here, feature selection can lead to improved classification. All of the classifiers that we studied—a generative Gaussian classifier, a discriminative logistic regression classifier, and a k-NN classifier, performed significantly better in the reduced feature space than in the full feature space.

We have not attempted to discuss the biological significance of the specific features that our algorithm identified, but it is worth noting that seven out of the fifteen best features identified by our algorithm are included in the set of 50 informative features used in (Golub et al., 1999), and moreover there is a similar degree of overlap with another recent study on this data set (Chow et al., in press). The fact that the overlap is less than perfect is likely due to the redundancy of the features in this data set. Note in particular that our algorithm works explicitly to eliminate redundant features, whereas the Golub and Chow methods do not.

We have compared feature selection to regularization methods, which leave the feature set intact, but shrink the numerical values of the parameters toward zero. Our results show that explicit feature selection yields classifiers that perform better than regularization methods. Given the other advantages associated with feature selection, including computational and interpretational, we feel that feature selection provides the preferred alternative on these data. It is worth noting, however, that these approaches are not mutually exclusive and it may be worthwhile to consider combinations.

References

Ben-Dor, A., Friedman, N., & Yakhini, Z. (2000). Scoring genes for relevance. *Agilent Technologies Technical Report AGL-2000-13.*

Bishop, C. M. (1995). *Neural Networks for Pattern Recognition.* Oxford: Oxford University Press.

Chow, M. L., Moler, E. J., & Mian, I. S. (in press). Identification marker genes in transcription profiling data using a mixture of feature relevance experts. *Physiological Genomics.*

Cover, T., & Thomas, J. (1991). *Elements of Information Theory.* New York: Wiley.

Dudoit, S., Fridlyand, J., & Speed, T. (2000). Comparison of discrimination methods for the classification of tumors using gene expression data. *Technical report 576, Department of Statistics, University of California, Berkeley.*

Golub, T., Slonim, D.K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J., Coller, H., Loh, M. L., Downing, J., Caligiuri, M., Bloomfield, C., & Lander, E. (1999). Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*, *286*, 531–537.

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

Koller, D., & Sahami, M. (1996). Toward optimal feature selection. *Proceedings of the Thirteenth International Conference on Machine Learning.*

Langley, P. (1994). Selection of relevant features in machine learning. *Proceedings of the AAAI Fall Symposium on Relevance.* AAAI Press.

Ng, A. (1998). On feature selection: Learning with exponentially many irrelevant features as training examples. *Proceedings of the Fifteenth International Conference on Machine Learning.*

Shalon, D., Smith, S. J., & Brown, P. O. (1996). A DNA microarray system for analyzing complex DNA samples using two-color fluorescent probe hybridization. *Genome Research*, *6(7)*, 639–45.

Tibshirani, R. (1995). Regression selection and shrinkage via the lasso. *Journal of the Royal Statistical Society B*, *1*, 267–288.

Xing, E. P., & Karp, R. M. (2001). Cliff: Clustering of high-dimensional microarray data via iterative feature filtering using normalized cuts. *Proceedings of the Nineteenth International Conference on Intelligent Systems for Molecular Biology.*