

A comparison study of Model based and Model free Linear Cancer classifiers

Yongxin Xi

Abstract

In this report we construct different types of supervised cancer classifiers and observe how their classification power evolves under a linearly growing number of gene features. We compare different classifiers based on the same feature selection mechanism. For filter classifiers, we tested on two datasets (Leukaemia cancer data, which does not have sufficient training samples; Yeast-cell data, which has sufficient training samples) and compared the model based such as FDA (Fisher Discriminant Analysis) classifier and several model-free such as WV (weighted voting), SVM (Support Vector Machine), IWV (Innovative Weighted Voting) classifiers. We observe that if there is not sufficient number of training samples, the model free classifiers will gradually outperform the model based classifier as the dimension of features goes up. Under sufficient training samples, however, the Fisher's classifier has a better chance to beat the other two. For wrapper classifiers, the model based classifier with insufficient training samples can achieve satisfying classification power by carefully selecting the features. These features show high independence of each other, therefore reveals that discriminative power of the feature alone is not the best criteria for feature selection. On the other hand, the discriminative power of the gene also plays an important role in feature selection, as shown by the degeneration of IWV classifiers compared to the WV classifiers.

Part I Data Description

Leukaemia Data—2 categories (insufficient training samples)

We use the public gene expression data downloadable from <http://www.broad.mit.edu/cgi-bin/cancer/datasets.cgi>. There are two datasets containing the initial (training, 38 samples, 27 ALL patients, 11 AML patients) and independent (testing, 34 samples, 20 ALL, 14 AML) datasets. These datasets contain measurements corresponding to ALL and AML samples from Bone Marrow and Peripheral Blood. For each patient, the data consists of 7129 un-normalized gene expressions. In addition we generalize two new datasets from the training data and testing data for further analysis: randomly select half of the training data and testing data and render it as the new training data, while the left data becomes the testing data (36 training patients: 24 ALL, 12 AML, 36 testing patients: 23 ALL, 13 AML)

Yeast Data—5 categories (sufficient training samples)

Each gene of the dataset has 17 features—expression levels at 17 different periods. There are 422 labeled genes, so we use these to test the classification methods. The five categories are S, M, late G1, G2 and early G1 genes. We divide the labeled genes into two datasets: training set and

testing set, and each consists of half data from every category. So there are in all 210 training samples and 212 testing samples.

Part II Description of classifiers

1. Feature selection:

To compare different classifiers, we adopt the same feature selection criteria, i.e. signed SNR, to each classifier.

2. Two main types of classifiers: Model-based vs. Model-free

Model-based classifier prototype:



Fig. 1

Model-free classifier prototype:

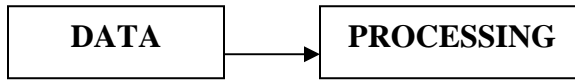


Fig. 2

1) Weighted Voting classifier (Model-free):

The advantage of WV classifier is its pellucid idea and ease of construction. It has been shown that classifiers perform better with not only positive SNR features but also negative ones. So after preprocessing of normalizing each gene, we divide all the genes into two groups: group A genes have higher expression for ALL patients, while group B genes higher for AML. For each group, select same number of genes (n) with highest SNR values from the two groups, and hence form the feature pool for the classifier ($2n$ features totally). The classification is based on a simple voting mechanism. For a new sample, each feature votes either for ALL or AML, and the associated weigh of voting is:

$$V^*(i) = SNR(i) \cdot |g(i) - u^*(i)|, \text{ for } i = 1, 2, \dots, 2n,$$

where $*$ stands for either ALL or AML.

$$V^{ALL} = \sum V^{ALL}(i), V^{AML} = \sum V^{AML}(i)$$

After collecting all the votes, a prediction is made by comparing the two voting scores (V^{ALL}, V^{AML}): the class which gets higher votes wins.

2) SVM (Support Vector Machine) (Model-free)

Support Vector Machine (SVM) seeks a separating hyper-plane that would maximize the distance (along the norm of the plane) of the supporting vectors of two classes. For the non-separable linear SVM, a fuzzy region is produced by the introduction of slack variables which gives an indication of classification error of each training sample. Consider the cancer classification problem as a non-separable linear SVM, and then the construction of SVM is essentially a quadratic programming problem:

$$\text{Min}_{w,b,\xi_i} \frac{1}{2} \|w\|^2 + C \sum_i \xi_i$$

$$\text{Subject to: } y_i(x_i^T w + b) \geq 1 - \xi_i, \forall i$$

$$\xi_i \geq 0, \forall i$$

Where w and b define the norm and intercept of the separating hyperplane respectively, ξ_i is the i th slack variable of the sample, x_i is the i th training sample, y_i is the label for the i th sample, and parameter C measures the weight of the penalty term in the objective.

The problem can be solved by MOSEK optimization package, by transforming into a standard form of QP:

$$\text{Min}_x \frac{1}{2} x^T Q x + c^T x$$

$$\text{Subject to: } l c \leq A^T x \leq u c$$

Where $x = (w_1 \ w_2 \ \dots \ w_m, \ b, \ \xi_1 \ \xi_2 \ \dots \ \xi_n)^T$

After optimization, $w^T x + b = 0$ defines the separating hyperplane for the two classes. For every new sample, if $w^T x + b > 0$, then predict ALL, otherwise classify the sample into AML patients.

3) FDA (Fisher's Discriminant Analysis) (Model-based)

Fisher's method looks for a hyperplane which separates the class means (u_1, u_2) well, while achieving a small variance of samples of each class (σ_1^2, σ_2^2) . In other words, the

objective of FDA is $\text{Min} \frac{(u_1 - u_2)^2}{\sigma_1^2 + \sigma_2^2}$.

FDA assumes that each class has a Gaussian distribution with the same variance, and therefore is a model-based classification method. Although we do not know the model, we estimate it by the current training data. Therefore only with a large number of the training samples can we estimate the model parameters well enough to generalize to new samples.

Denote the two class means as c_1 and c_2 , then we have the estimated variance S_w :

$$S_w = S_1 + S_2$$

$$= \frac{1}{N_1} \sum_{i: y_i=1} (x_i - c_1)(x_i - c_1)^T + \frac{1}{N_2} \sum_{i: y_i=-1} (x_i - c_2)(x_i - c_2)^T$$

$$w_{opt} = S_w^{-1}(c_1 - c_2)$$

$$b = -\frac{1}{2} w^T (c_1 + c_2)$$

w_{opt} is the norm to the optimal separating hyperplane, b is the intercept, and they define the separating hyperplane of the problem.

3. K-category classification

Suppose now there are $K > 2$ classification problems we wish to solve. The classification methods popular in practical use are OVA (one against all), OVO (one against one), which was introduced by Vapnik in 1998 and Krebel in 1999, respectively. In OVA, K binary classifiers will be constructed, each from which is a classifier for one category against all the rest categories.

In OVO, we construct $\binom{K}{2}$ binary classifiers for all pairs of different categories. These two methods have been shown to achieve similar performances (Support Vector Machine for Multi-class Pattern Recognition, J. Weston and C. Watkins), but OVA provides an easier and neater problem formulation. So we use OVA in our K-category classification study.

Part III Experiments

A. Classification Accuracy of Leukaemia data (insufficient training sample)

A.1 Three testing methodologies

We tested on the Leukaemia data with three classifiers by three methods: 1) Leave one out 2) Independent testing 3) Newly generalized data. And observe their classification performance under a linearly growing number of selected features.

1) In leave-one-out testing, only the training dataset (38 samples) are used. At each time, one sample from 38 samples is left out and the classifier is trained on the remaining 37 samples then test on the left-out sample. Since there are only 37 training samples at each time, the maximum number of pairs of features for the Fisher's classifier is 18. We therefore compare the classification performance based on 2 to 36 features.

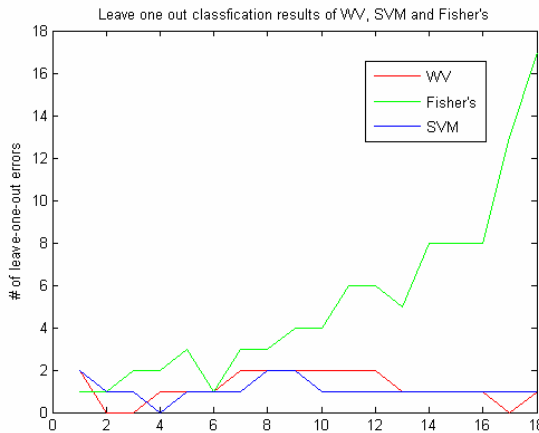


Fig. 3

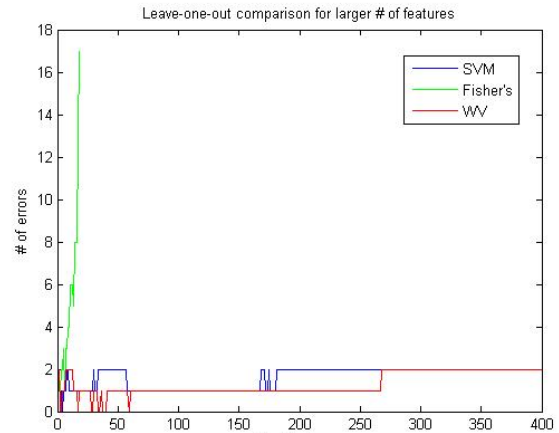


Fig.4

In Fig.3 and Fig. 4 the X-axis stands for the number of pairs of feature used, and Y-axis stands for the number of classification errors among 38 leave-one-out samples. We observe that Fisher's classifier has an increasing probability of classification error as the feature number goes up. Since the maximum possible number of errors is 37, the classification performance of Fisher's classifier using 18 pairs of features (i.e., 36 features in all) is just above the chance. On the other hand, both WV and SVM have rather good classification accuracy consistently with number of features ranging from 2 to 800.

2) In independent testing, classifiers are constructed by the 38 training samples and tested on the 34 independent testing samples (20 ALL, 14 AML). Fig.5 illustrates the classification accuracy of WV (red), SVM (blue) and Fisher's (green). Again Fisher's classifier is close to WV and SVM when number of features is small but tends to perform

worse as it goes up. Below 18 pairs of features, performances of WV and SVM are similar to each other. However, the red and blue curves in Fig.6 show that SVM outperforms WV from 200 pairs of features and above.

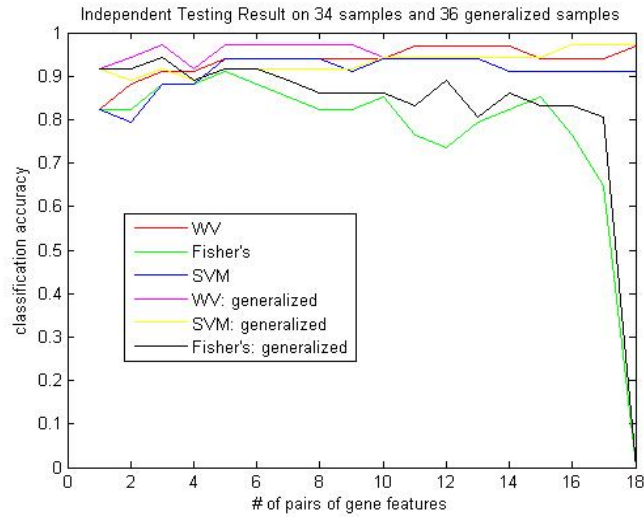


Fig. 5

3) Before reaching any hasty conclusions, we generalize a new dataset and test these two classifiers under large number of features. And this time as shown in Fig. 6, these two methods (SVM, WV) achieve similar performances. Because the generalized training set is drawn from half of the original training set and half of the testing set, the distributions of generalized training set and generalized testing set are more identical than the original ones. Therefore it is very likely that the WV method is not as good as the SVM method when statistical distributions of the testing data are not similar enough to those of the training data.

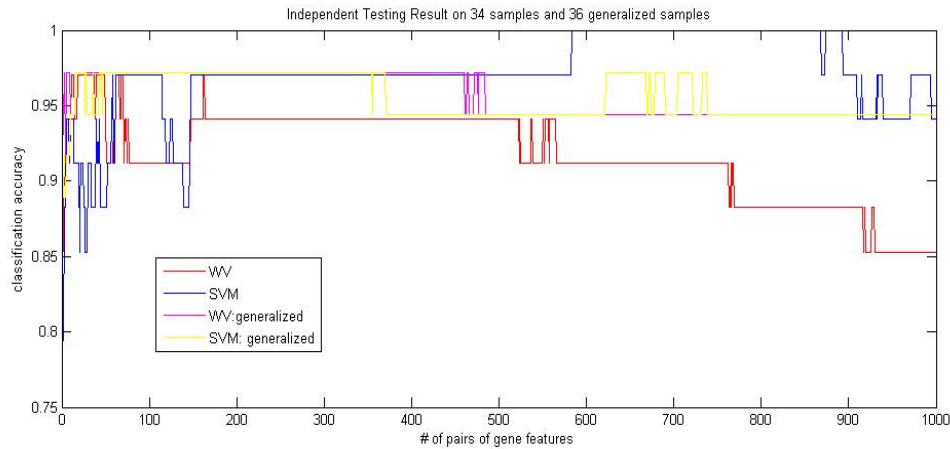


Fig. 6

A.2 Discussion

There are several reasons why Fisher's classifier does not work so well for our data.

1. Model based: sometimes our data does not apply to the model
2. Fisher's classifier suffers from the lack of numerical pool of examples
3. Feature selection problem: Wrapper classifier needed

1. Fisher's classifier assumes the data to be Gaussian distributed and different categories of data share the same "distribution shape"—the same covariance matrix. Theoretically, if we know the statistical model governing the data generation process, the model-based classifier could be most relevant to the data and render best classification. Practically, we almost always lack the underlying statistical. Although we can assume that the training data and the testing data are generated from the very same statistical distribution, it will cast doubt on most statistical based criteria and methods. Also, for most data mining and pattern recognition applications in bioinformatics, a prior model of the data cannot be assumed.

2. While Fisher is built upon the underlying Gaussian distribution, SVM seems to aim at better separating the data from two distinct classes. Often the availability of reliable and representative training data is also of critical concern. Training data size differs greatly from one application to another (such as the two datasets discussed here).

In many applications, we are fortunate to have large sets of training data available. Statistical analysis provides an effective mathematical manipulation of these data. It helps model the data and find hidden relationship among them. If the number of training examples is too small, then it becomes imperative to rely on a valid and effective assumption on the underlying statistical model which governs the creation of the finite training patterns. If the number of training examples is too small or the number of features is too large, it may possess little prediction or generalization capability.

3. Another possibility of Fisher's poor performance is the inappropriate feature selection. Fisher classifier itself can cope with the redundancy of features, yet the feature selection scheme we adopt here might be highly dependent of each other and hence the combination of them does not provide much more information than the original features. To see that with careful feature selection, Fisher's classifier could reach very high prediction accuracy, which will be shown in the following section.

B. Classification Accuracy of Yeast-cell data (sufficient training sample)

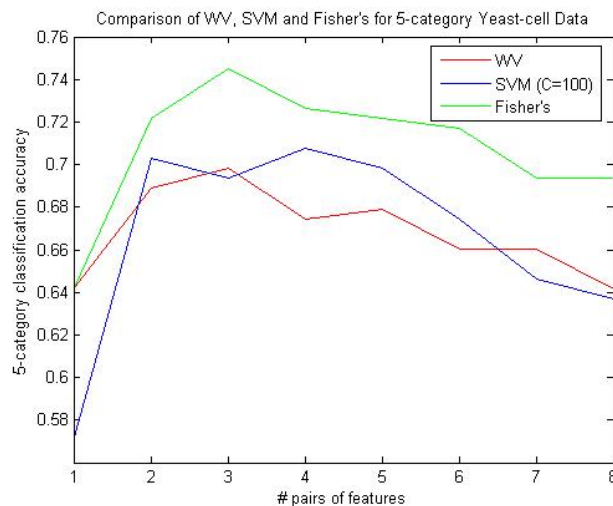


Fig. 7

Unlike the Leukaemia data, the features in Yeast-cell data are the temporal information of expression levels, and the categories are the genes themselves. A major difference of this dataset from the Leukaemia data is a much smaller feature space. There are in all 17 features which could be used in classification, while there are over 200 training sample available. In other words, the training samples are sufficient for the Fisher's classifier to generalize a reasonable model for this case. From Fig. 7, we can see that Fisher's classifier does a much better job than it did in Leukaemia data—comparing with WV and SVM. (The scheme of feature selection is the same for these three classifiers.)

Also, when the number of features changes from 2 to 16 (maximum 17), the performance curves of all types of classifiers reveal an approximately concave waveform. It shows the importance of feature selection, because using all of the features usually does not give us a satisfying classifier.

Fig. 8 shows performances of SVM classifiers using different weight parameter C. The similar waveforms suggest that SVM is not very sensitive to this parameter.

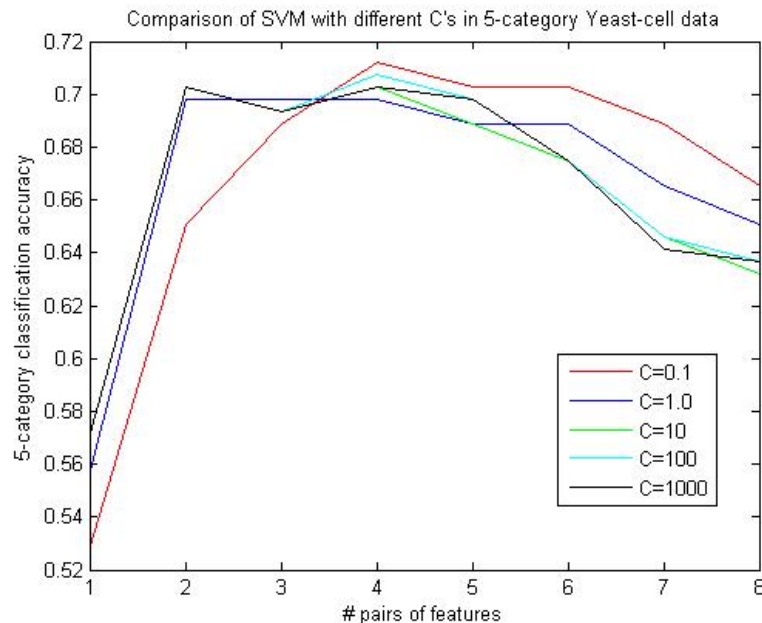


Fig. 8

C. Wrapper Classifier

Fig. 9 and Fig. 10 are the overall structure of filter method and wrapper method for classification. What distinguishes wrapper classifiers is that it uses classification result as a feedback guidance to update the selected features.

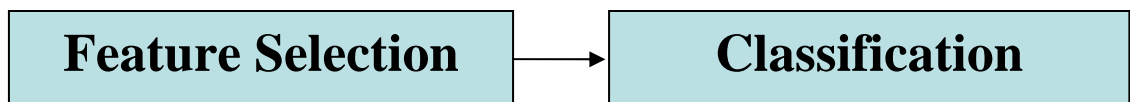


Fig.9 Filter Classifiers

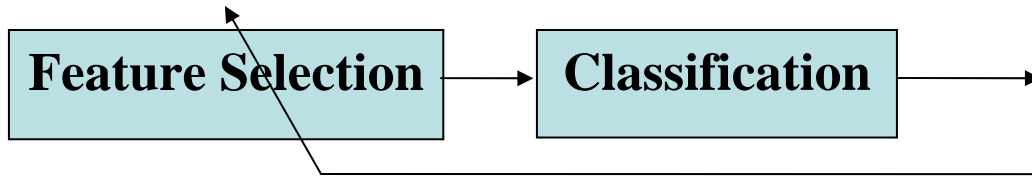


Fig.10 Wrapper Classifiers

If we would like to construct a three featured wrapper Fisher's classifier, the simplest yet impractical way to do this is to construct all the classifiers using $\frac{n!}{(n-3)!3!}$ combinations

of three features and search for the one which gives best classification result. Since n (the number of genes in our data set) equals 7129, such thorough search would require 6×10^{10} classifiers to be constructed!

To tackle this problem, we use the forward search method to increase the number of features one by one.

1. One-feature case

Search the whole gene pool to find one gene which gives best classification result. We find that gene # 4653 or gene #4847 achieves 91.66% classification accuracy (3 errors /36). To see the property of the genes, we look into their SNR values. Gene # 4847 has third largest SNR among group 2 genes (those who express higher in AML patients), and gene # 4653's SNR ranks No.368 among group 2. Therefore, good gene for wrapper classifier does not necessarily have high SNR value.

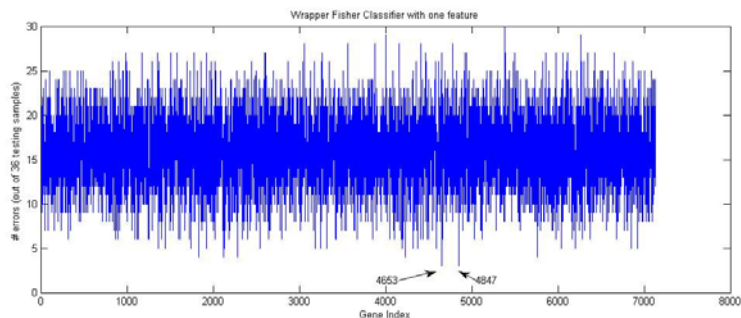


Fig. 11 Classification errors of one feature Fisher classifiers

2. Two-feature case

Fix the first feature to be gene #4847, and select the second feature. The combinations of (#4847, #804), (#4847, #1796), (#4847, #1941) or (#4847, #6477) all reaches 100% classification accuracy. Further analysis of these genes show they are P11, N1095, N516, P123 (P means the gene group with higher expression in ALL, while N in AML; the number following P or N means the rank of SNR of the gene).

Fig. 12 shows the classification results of Fisher's classifiers using #4847 as the first feature and any other gene as the second. What is remarkable in the figure is most 2-feature fisher classifiers have only 2 or 3 errors.

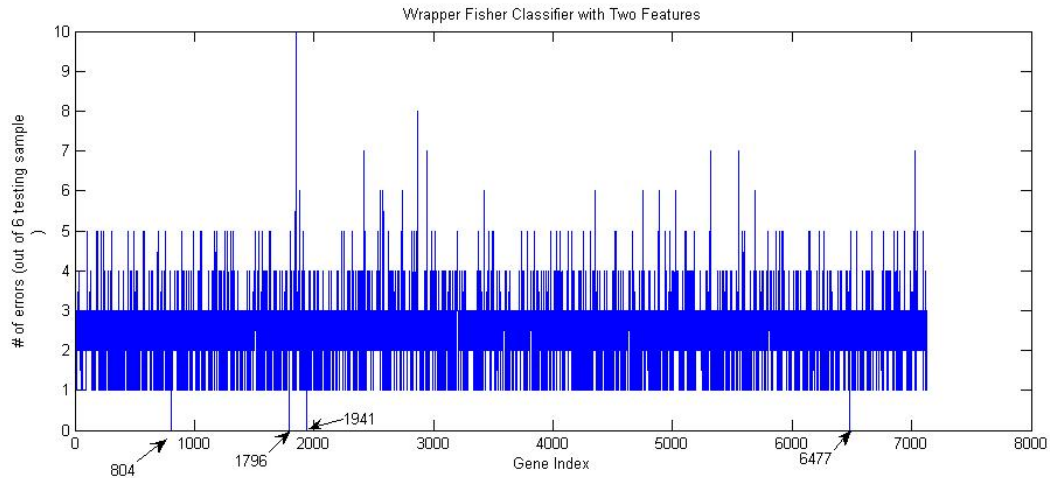


Fig. 12 Classification errors of two-feature Fisher classifiers (with first feature fixed)

3. Three-feature case

Fix the first two features to be gene #4847 (N3) and #804 (P11), and test Fisher's classifiers with any other third feature. Now most classifiers have either 0 or 1 classification error.

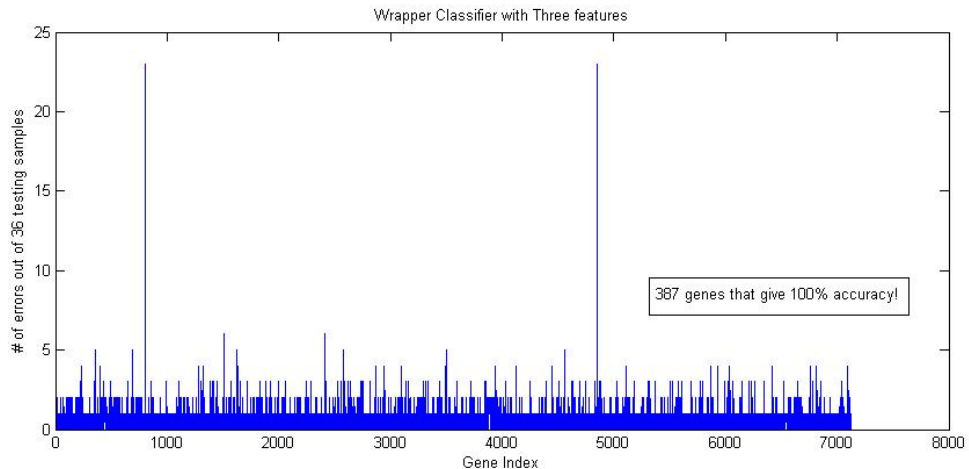


Fig. 13 Classification errors of three-feature Fisher classifiers (with first feature fixed)

4. Innovation analysis

By now we have shown that with careful choice of features, wrapper fisher classifiers can easily reach 100% classification accuracy.

It would be interesting to look at what kind of combination of genes performs best in classification. Although gene 4847 and gene 4653 both rank first in the single feature selection phase, the combination does not come first at double feature phase. Instead, gene 4847 and gene 804 ranks first with 100% accuracy. The innovation analysis of these genes measures the dependence of gene pairs, and gives an explanation for this. The innovative power of #4653 based on #4847 is 0.5214 while the innovative power of #804 based on #4847 is 1.4128, which is about three times higher. This implies gene 4653

shows more dependency on gene 4847 than gene 804, and therefore provides less information if used as the second feature.

5. IWV

However, independence alone is also not enough for good feature selection, because of lack in the discriminative power. Fig. 14 shows that the Independent Weighted Voting classifier only reaches an average 70% of classification accuracy. Note that in IWV we only consider the independence during feature selection. (Fig. 15 shows the performance of WV using different number of features)

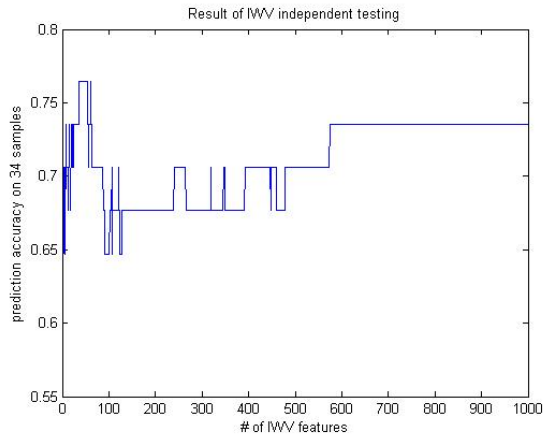


Fig. 14

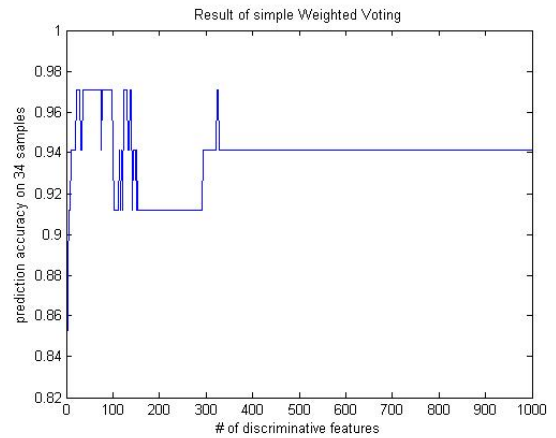


Fig. 15