Two-Phase EA/*k*-NN for Feature Selection and Classification in Cancer Microarray Datasets

Thorhildur Juliusdottir[1,2], Ed Keedwell[1], David Corne[1], Ajit Narayanan[3]

[1]SECAM, Harrison Building
Harrison Building
University of Exeter, UK

[2]School of Biological Sciences
Washington Singer Building
University of Exeter, UK

[3]School of Computer Science
Main Building
University of Portsmouth, UK

***Abstract*-Efficient and reliable methods that can find a small sample of informative genes amongst thousands are of great importance. In this area, much research is investigating the combination of advanced search strategies (to find subsets of features), and classification methods. We investigate a simple evolutionary algorithm/classifier combination on two microarray cancer datasets, where this combination is applied twice – once for feature selection, and once for further selection and classification. Our contribution are: (further) demonstration that a *simple* EA/classifier combination is capable of good feature discovery and classification performance *with no initial dimensionality reduction*; demonstration that a simple *repeated* EA/*k*-NN approach is capable of competitive or better performance than methods using more sophisticated pre-processing and classifer methods; new and challenging results on two public datasets with clear explanation of experimental setup; review material on the EA/*k*NN area; and specific identification of genes that our work suggests are significant regarding colon cancer and prostate cancer.**

I. Introduction

Microarray technology was first introduced by Pease *et. al.* [1], and has become vital in profiling gene expression patterns. Microarray density has significantly increased since 1994 and currently allows many thousands of genes to be assayed simultaneously. This vast amount of data leads to great statistical and analytical challenges.

Comparison of gene expression data between different samples (e.g. disease versus normal) can provide us with a deeper understanding of the disease and its development. Several gene expression profiles obtained from tumours such as colon [2], leukaemia [3], breast [4] and lymphoma [5] have been studied and compared to expression profiles of normal tissue. Such comparisons point towards genes that appear to be differently expressed in cancer and normal samples, which in turn fuels hypotheses towards deeper understanding of this complex disease and assists in drug discovery and early diagnosis. However, expression data are highly redundant and noisy and most genes are (believed to be) uninformative with respect to the classes studied. Only a fraction of genes may present distinct profiles for different classes of samples. Tools which can deal with these issues are critically important, so that we can learn to robustly identify a subset of informative genes embedded in a large dataset that is contaminated with high-dimensional noise [6].

'Feature selection' (FS) methods, in this context, search for informative sets of genes whose expression patterns are then used by a classification method to distinguish between different classes of samples. A number of heuristic approaches have been used for FS including sequential forward selection, branch and bound, and evolutionary algorithms [7, 8, 9] and many more recent works. It is becoming popular to use sophisticated nonlinear classifiers for the `classification' phase, however we wish to put forward the argument that, in the combined feature selection/classification context, it is highly valuable to focus on methods in which the classification method is straightforward (e.g. *k*-NN, or a linear discriminant function), since this places a greater onus on the search method to find salient and significant gene subsets. In turn, this leads to gene *selection* results which are arguably more valuable in the context of gene targeting. In other words, the use of a highly competent nonlinear classification tool (e.g. an SVM or a multilayer perceptron) may skew the results in terms of the degree to which they point to significant genes, since the capability of the classifier can potentially cloud the effects of non-ideal `discovered' gene subsets.

Here we therefore focus on the use of an EA in combination with *k*-NN. In section II we provide a brief review, focussing on FS methods that combine EAs with *k*-NN. In section III we describe our methods and experimental setup. In section IV we describe the datasets used in this paper and our experiments, as well as discuss previously reported results with these datasets. In section V we describe results. We conclude in section VI.

II. Brief Review of Related Work

The aim of feature selection (FS) is to identify a minimum set of non-redundant features that are useful for classification, attempting to exclude irrelevant, noisy and redundant features. When analysing microarray datasets, FS may help in providing a deeper understanding of the molecular basis of cancer and assist in drug discovery and early diagnosis. Also, Guyon *et. al.* [10] state that the numbers of features to measure need to be drastically reduced in order to reduce costs in clinical settings.

The number of possible feature subsets grows exponentially with the number of features [11], making enumerative search infeasible. The only practical solutions

0-7803-9387-2/05/$20.00 ©2005 IEEE

are heuristic approaches, an example being the work of Raymer *et. al.* [12]. Another aspect is feature correlations; most previous FS work ranks features without considering this aspect [13, 14]. A benefit of approaches that use an EA to search the space of feature subsets is that consideration of these inter-correlations is implicitly built into the search strategy.

FS methods broadly fall into three types; the *wrapper* approach, the *filter* approach and the *hybrid* approach [11]. The wrapper approach seeks the best feature subset for use with a specific algorithm and the filter approach attempts to assess the merits of features from the data alone [15]. The hybrid approach attempts to do both, by exploiting their different evaluation criteria in different search stages [11].

Evolutionary algorithms [16, 17, 18, 19] have been successfully applied to a broad spectrum of optimization problems, including many pattern recognition and classification tasks. When an EA is applied to gene expression data, each individual (called a *chromosome*) represents a specific set of genes, which constitutes a candidate solution to the discrimination problem [6]. The fitness of such a chromosome is usually the accuracy with which a given classification method can classify the data using only the features specified in the chromosome.

Meanwhile, the *k*-NN classifier, introduced by Fix & Hodges [20], is a fast and simple classifier that is easy to implement. It has been been demonstrated to have good classification performance on a wide range of real-world datasets [21]. The basic idea is to assign a class label to a test sample based upon the classes of the *k* most similar training samples [22]. Because of its speed, good classification performance and simplicity, *k*-NN is well suited as a classifier within the fitness function of an EA.

The basic idea of the hybrid EA/*k*-NN approach was (to our knowledge) first given in Siedlecki & Sklansky [7], which used the method illustrated in Figure 1. Here, we pretend that the data comprise only three samples, each containing six features. (a) The `original' data – each row is a sample consisting of features, where the class value (a or b in this case) is shown in the last column. (b) The EA uses a straightforward binary chromosome – in this case, the chromosome encodes the subset of features comprising features 1, 3 and 6. (c) Here we see the features extracted from the data for each sample, as directed by the encoded subset; (d) Finally, the chromosome is evaluated by running *k*-NN on the data using the extracted subset of features and obtaining the accuracy of classification on the training set. Each resulting subset is evaluated according to subset size and classification accuracy on a set of testing data using a *k*-NN. A small subset (with a small collection of "1"s) will score well. Feature subsets that include too many features and/or obtain poor classification performance get a poor fitness value and are unlikely to survive to the next generation.

The findings of Siedlecki and Sklansky [7] were that the time needed for finding near-optimal subsets of features from large datasets could be much reduced by applying an EA in combination with *k*-NN. This has inspired many to extend and modify this basic idea and apply it to a range of other classification problems [6, 8, 9, 23].

Of importance here, however, is work by Jirapech-Umpai and Aitken [24] who used the RankGene software for FS. Initially they applied their EA/*k*-NN method, without prior FS, to Golub's leukemia dataset [3] which contains 7070 genes. The initial number of randomly selected features in a chromosome was set to 10, and they used small population sizes. Results show poor accuracy on the test set (68% at best), and their EA typically converged quickly. (with such a large gene set and a small chromosome size, the risk of getting stuck in local optima is high. It would have been interesting to change the termination criteria of the EA and let the algorithm run for much longer.) However, accuracy improved significantly when Rankgene was used for FS prior to classification. They found the 100 best genes by using the RankGene. Using the EA/*k*-NN only on these resulted in test set accuracy of 95%.

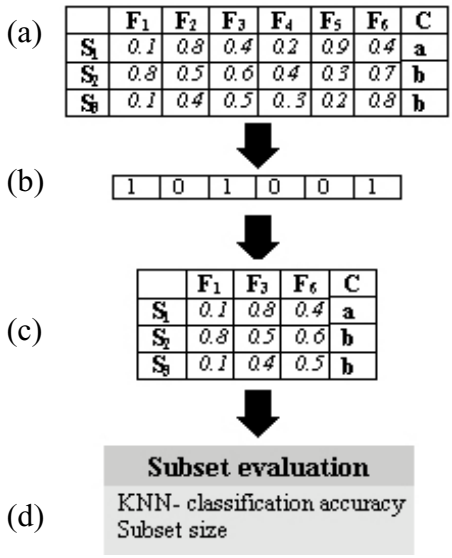

Figure 1. A simple example of how features are filtered out from the original dataset using a chromosome with a binary encoding. The features that are filtered out are used for classification. The original data set only consists of 6 features and 3 samples for clarification. See text in Section II for explanation.

It is also very well known that stochastic search methods, especially when applied to truly complex and difficult problems, will return different results in different runs. In the FS context (which has already appeared in papers we have discussed), this ends up being of benefit, since we can analyse the genes found in terms of their frequency of occurrence in the best subsets of repeated runs. We can have some degree of confidence that very frequently occurring

genes are significant in terms of the disease (or other mechanism) under study.

Our basic idea in this work is to focus on EA/*k*-NN as the method, and to accommodate the findings that prior selection is good by *also* using EA/*k*-NN for this (the prior FS) step. We also exploit the fact that repeated-run analysis can yield significant results, and do this by obtaining subsets of frequently-appearing genes and again testing their *k*-NN accuracy. In using *k*-NN throughout we avoid a more sophisticated classifier (than *k*-NN) that might otherwise achieve good performance in spite of non-ideal significance in the gene subset.

III. Methods

Our hybrid EA/KNN approach, uses the EA as a stochastic search algorithm and *k*-NN as a classifier. The overall approach is explained in figure 1. All experiments herein were carried out using *Matlab* and the *Genetic Algorithm and Direct Search (GADS)* toolbox. We describe next the chromosome representation, the fitness evaluation method and the mutation method; for readers also intending to use the GADS toolbox, we provide helpful detail on how these were implemented into the GADS toolbox.

Our encoding differed from the binary encoding referred to figure 1. We chose instead to explicitly represent a subset of features as variable length (up to a maximum) list of integers, each pointing directly to a specific feature. Hence, the list ``23, 55, 1281'' would encode the subset comprising features (genes) 23, 55 and 1281. Integers in this list were allowed to range from 0 to *ngenes*, where *ngenes* is simply the total number of genes in the dataset. A `0' indicated `no' gene. The initial chromosome only contains non-zero values. This encoding had the benefit of limiting *a priori* the size of a feature subset, and the related benefit of scalability – i.e. when dealing with microarray datasets with many thousands of features, the binary chromosome of figure 1 would need to contain many thousands of bits.

The fitness function includes the *k*-NN classifier. In all experiments reported in this paper, we use the value k=3, which was determined from preliminary experiments. These experiments involved testing, on one of the datasets, values of k from 2 to 10 in steps of 1, in order to ascertain if one of these values of k provided consistently fair classification results. Three-fold cross-validation was used in all cases. Fitness was calculated by using the following function (expressed here in Matlab syntax):

```
fitness=((100-class_acc)/100)+((n/N)/α);
```

where *class_acc* gives the mean classification performance over the three three-fold cross-validation runs, n is the size of the gene subset encoded by this chromosome, N is the maximum chromosome length (the largest a feature subset could be), and α is a parameter controlling the tradeoff between preference for accuracy and preference for small subset sizes.

The mutation rate was set to 0.3 for all experiments. When a chromosome was selected for mutation, there was a 30% chance of mutating a randomly chosen gene into a randomly chosen non-zero value, and a 70% chance of mutating a gene to 0 (i.e. removing it from the encoded subset).

For each dataset, we investigated a two-phase approach. In phase A, the EA/*k*-NN hybrid was run on the raw data, finding and evaluating subsets of the entire collection of genes for that dataset. We recorded the results from this phase, which are indicative of the performance of a `basic' EA/*k*-NN approach. However, we also collected those genes that appeared in the final populations of repeated phase A experiments, resulting in a large yet much reduced subset of the genes in the original data. In phase B, we simply ran the EA/*k*-NN approach (otherwise unaltered), on this reduced subset. In other words, phase B represents a normal EA/k-NN run, but which makes use of phase A as an *a priori* FS method. We record results from both phases, as well as the results of selected investigations of the various gene subsets found. This is all described in more detail below.

IV. Experiments

Experiments were carried out on two cancer-related microarray datasets: colon [2] and prostate [25]. Initially a number of runs were carried out in order to find suitable parameters. Each dataset was divided into training/testing samples so that 75% of the data was used for training/testing and 25% of the data was used for validation. The relative proportions of the two classes in the dataset (cancer, non-cancer) was kept as equal as possible in each set. Accuracy was calculated by using the standard 3-fold cross validation procedure.

Two experiments (A and B) were carried out on each of the datasets. The colon cancer dataset was shuffled between experiments in order to get a reasonable distribution of samples in the training/test and validation datasets. The prostate cancer dataset did not need to be shuffled.

A. Experiments 1A and 1B: prostate cancer

The prostate dataset [25] involves 12600 genes, derived from 52 prostate cancer samples and 50 normal samples. The training set contained 75 samples (the first 39 cancer and the first 36 normal samples). This was further divided into `folds' for 3-fold cross validation, each containing 25 samples (13 cancer and 12 normal). The remaining 27 samples were used as the validation set.

B. Experiment 1A (*EA/*k*-NN on all 12600 genes*)

The EA/*k*-NN algorithm was run ten times using the whole dataset. The EA settings were: 400 generations, chromosome length 400, population size 80, elite count of 2 (the best 2 chromosomes from a generation were always copied directly into the next generation), roulette wheel selection and single-point crossover, where each crossover resulted in the generation of two children. The k value for the *k*-NN

classifier was 3, as determined from preliminary experimentation. The mean 3-fold cross-validation accuracies from these ten runs are listed in Table I. Final best subsets from each run varied in size between 20 and 48. These were pooled to obtain a new larger subset which only contained the best solutions. This contained 245 unique genes.

C. Experiment 1B (*EA/*k-*NN run on 245 genes*)

Experiment 1B ran the EA/*k*-NN using only the 245 genes of the final subset from experiment 1A. The parameters were as in experiment 1A, except for generations (100), chromosome length (100), and population size (30). Ten runs were carried out, resulting in gene subset sizes ranging from 7-11. These were pooled to form a subset of size 20. We noted that three of these 20 were in eight or more of the original subsets. These were further analysed.

D. Experiments 2A and 2B: colon cancer

The colon cancer dataset contains 2000 features and 62 samples (40 cancer and 22 normal) [2]. The dataset was shuffled and then divided into four subsets of similar sizes. Three, of the subsets (45 samples) were used for the 3-fold cross-validation as training and testing samples (train on two samples, test on one). The fourth subset (10 cancer, 7 normal)) was used as the validation set. The settings were: chromosome length 200, population size 30, single point cross-over, 500 generations, roulette wheel selection, elite count of 2, and mutation rate 0.3.

E. Experiment 2A (EA/k-NN on all 2000 genes)

The EA/*k*-NN was run ten times on the original colon cancer dataset. The best final subsets of genes from the corresponding runs were combined, yielding a set of 151 unique genes. All parameters apart from two were kept the same as for the prostate cancer data. Since the colon cancer dataset is much smaller. the chromosome size was cut to 200 and the population size was cut to 30. The *k* value for the *k*-NN classifier was kept constant as 3 for all runs.

F. Experiment 2B (*EA/*k-*NN run on 151 genes*)

Experiment 2B ran the EA/*k*-NN using only the 151 genes that were included in the final subset from experiment 2A. Two parameters from experiment 2A were modified: chromosome length (70), and number of generations (100). Eight runs were carried out. A new subset was formed from the resulting final subsets, which contained 37 genes. Some additional experiments were done including the most frequently selected genes derived from the eight runs.

G. *Comparative results on the prostate dataset*

Before we present and discuss the results, we briefly review here the findings obtained so far in other research using these datasets (mainly using more sophisticated classifer methods than *k*-NN).

Topon and Iba [26] applied a Probabilistic Model Building Genetic Algorithm (PMBGA), with Support Vector Machine as a classifier, to the prostate cancer dataset, after a preprocessing method to the dataset which reduced the number of genes in the dataset to 5966. They calculated the training accuracy by using Leave-One-Out-Cross validation (LOOCV). The dataset was normalized and divided into training and test sets, each containing 50% of the total samples. They collected 177 different subsets. Their best gene subset returned a test set accuracy of 94,12%, including 24 genes. The smallest subset that they found contains only 6 genes that returned a 82.35% testing accuracy.

The average test set accuracy returned directly by RPMBGA (using 50% of the samples for training and 50% for testing) is 84.29+/-4.57 with the average number of selected genes being 17.14+/-7.40.

Singh *et. al*. [25] combined signal/noise statistics and the *k*-Nearest Neighbor classifier. This enabled them to find a 16-gene subset that returned 93.12% *training* accuracy using *all* 102 samples. Singh *et. al*. [25] say that models that utilized 4 or more genes classified samples with >90% accuracy.

We note that both sources found so far which provide results on the prostate data set are problematic in terms of direct comparison. The Singh *et. al.* [25] experiments used *all* the data for training, but we prefer to use the more favoured approach in which test accuracy on a previously unseen validation set gives a better estimate of performance on unseen data. Meanwhile, Topon & Iba [26] use a 50/50 train/test split, but we prefer manifold cross-validation since it provides more reliable estimates of performance on unseen data. Tentative and qualified comparisons of our results with these are nevertheless possible.

H. Comparative results on the colon dataset

Li. *et. al.* [23] divided the total of 62 samples in the colon dataset into 42 training (the 42 first samples) and 20 testing samples. They ran an EA multiple times using 3-NN as the classifier in the fitness function, obtaining in all a total of 6388 subsets containing 50 genes each. The frequency of each gene was calculated and a ranked list of the most frequently selected genes was formed. The top ranked genes were used to form the "best subsets" of genes. Using this method, their best test accuracy for such a 50-gene subset was 65%.

Liu *et. al.* [27] obtained a classification accuracy of 91.94**%** on the colon cancer dataset using their LOOCV method. They state "*85.48% predictive accuracy is the best classification result obtained in Dettling et al. (2003)* [28]*, where they used various boosting algorithms and adopted leave-one-out cross validation (LOOCV)*." Again, the results are not compatible with our experimental design – we have eschewed LOOCV owing to its computational complexity on large datasets (at least in terms of features), and we prefer more manifold cross-validation than used by many other researchers. However, we note again that qualified comparisons can be made.

V. RESULTS

A. Prostate cancer: Results and Analysis

Results from experiments 1A and 1B are in Table I. Ten subsets including an average number of 28 genes, were found by applying the EA/*k*-NN method to the original 12600 genes. The average test set accuracy was 87.04%. This improved to 88.88% when the EA/*k*-NN was applied in two phases, in which the first phase was used to reduce the features down from 1260 to of 245 genes. The number of selected genes was reduced to an average of 9 genes in each subset in the second phase. The union of all these subsets yielded a set of 20 unique genes. The classification accuracy on the validation set using these 20 genes was 88.88%. This is slightly better than the best of Topon and Iba's results using RPMBGA, but not as good as their results using LOOCV, and also not as good as Singh et al's results. However, as indicated, none of these results is comparable. But, as a very rough rule, all else being similar, we could expect to achieve better (worse) performance than a previously reported method if the previous method used smaller (larger) training set sizes. In the Topon and Iba, and Singh et al, comparisons, we would therefore expect worse performance in each case, so the comparison against RPMBGA is promising with respect to the potential for our approach.

In the ten subsets generated from experiment 1B, three of the genes were more frequently selected than the other 17. These were: *31444_s_at* (9 of 10 subsets), *35905_s_at* (8 of 10) and *216_at* (8 of 10). Table II also shows the individual performance of each of these genes alone (i.e. as a singleton subset) on the validation set. Performance using different combinations of these three genes is given in Table III.

Genes *35905_s_at* and *216_at* return the best classification of 88.89% accuracy, regardless of whether gene *31444_s_at* is included. This accuracy is competetive with other test accuracies reported for these data, and is remarkable in arising from a subset of just 2 genes. Meanwhile, gene *216_at* has the highest individual classification accuracy of 77.78%.

TABLE I. THE AVERAGE TRAINING AND TEST ACCURACY AS WELL AS THE AVERAGE NUMBER OF GENES DERIVED FROM EXPERIMENTS 1A AND 1B RESPECTIVELY. THE AVERAGE VALUES ARE CALCULATED OVER THE RESULTS OBTAINED FROM TEN GA/KNN RUNS ON THE ORIGINAL SUBSET OF 12600 GENES AND THE DERIVED SUBSET OF 245 GENES RESPECTIVELY. THE TRAINING ACCURACY WAS OBTAINED BY APPLYING GA/KNN WITH 3-FOLD CROSS-VALIDATION AND THE TEST ACCURACY WAS MEASURED BY USING KNN TO CLASSIFY THE VALIDATION SAMPLES .

	Expt 1A	Expt 1B
Avg training accuracy /Std:	97.2000 +/- 1.1675	98.66 +/- 0
Avg test accuracy /Std:	87.0370 +/- 4.3649	88.88 +/- 0
Avg number of genes /Std:	27.6000 +/- 8.3560	8.6000 +/- 1.8379

TABLE II. A RANKED LIST OF THE GENES WHICH WERE THE MOST FREQUENTLY SELECTED IN THE FINAL SUBSETS OF EXPERIMENT 1B. THE CLASSIFICATION ACCURACY USING ONE OF THE THREE GENES SINGLY, IS ILLUSTRATED IN THE THIRD COLUMN OF THE TABLE.

Rank	Gene	Test Acc.
1	3144_s	74.0741 %
2	216_at	77.7778 %
3	35905_s_t	59.2593 %

TABLE III. THE TABLE SHOWS THE TEST ACCURACIES THAT WERE OBTAINED WHEN THE VALIDATION SET WAS CLASSIFIED BY *K*-NN USING DIFFERENT SUBSET VARIANTS INCLUDING THE THREE MOST FREQUENTLY SELECTED GENES. THE CLASSIFICATION ACCURACY USING ALL THREE GENES IS ILLUSTRATED, AS WELL AS THE CLASSIFICATION ACCURACY USING TWO OUT OF THREE GENES.

Subset of genes:	Test Accuracy
31444_s_at, 35905_s_at , 216_at	88.8889 %
31444_s_at , 35905_s_a	66.6667 %
31444_s_at , 216_at	85.1852 %
35905_s_at, 216_at	88.8889 %

B. Brief Analysis of Selected Genes

Gene 216_at is *prostaglandin D2 synthase* which has been identified as differentially expressed in androgen ablation-resistant prostate cancer [29]. This `androgen ablation' therapy is very common in the treatment of prostate cancer. However, virtually all prostate cancers respond to this but eventually develop resistance. Holzbeierlein et. al. [29] identified 645 differentially expressed genes in patients undergoing treatment and patients that had developed resistance, by applying a hierarchical clustering method to a

dataset consisting of 63,175 probes. The prostaglandin D2 synthase was amongst these 645 which are believed to play a part in the mechanisms of androgen ablation therapy resistance.

Gene 31444_s_at (also present in Topon & Iba's final subset of 17 genes) is Annexin II (lipocortin II). The expression of Annexin II has been found to be lost or reduced in prostate cancer. This may contribute to the development and progression of prostate cancer [30].

Gene 35905_s_at is a glyceraldehyde-3-phosphate dehydrogenase (GAPDH). In the link: -- http://www.researchd.com/miscabs/trk5g4.htm -- we find the quote: "*During the last decade, many findings have been made concerning the role of GAPDH in different pathologies including prostate cancer progression*". Epner & Coffey [31] concluded that multiple forms of GAPDH might play diverse roles in normal prostate tissue and in prostate cancer.

C. Colon cancer: Results and Analysis

The average results from eight runs in experiments 2A and 2B are in Table IV. When the 3-NN classifier was tested on the 151 genes collected from phase 1, it could classify 14 out of 17 (82.35%) samples in the validation set. When the subset of 151 genes was used as the initial gene pool for experiment 2B, the average taccuracies over eight runs again improved, and final gene subsets were again significantly reduced.

TABLE IV. THE AVERAGE TRAINING ACCURACY WAS CALCULATED OVER EIGHT RUNS OF APPLYING EA/*K*NN WITH 3-FOLD CROSS-VALIDATION TO THE TRAINING DATA (45 SAMPLES). THE ORIGINAL GENE POOL IN EXPERIMENT 2A INCLUDED ALL 2000 GENES IN THE DATASET. THE ORIGINAL GENE POOL FOR EXPERIMENT 2B CONSISTED OF A SET OF 151 GENES WHICH WERE DERIVED FROM EXPERIMENT 2A. THE TEST ACCURACY WAS MEASURED BY APPLYING 3-NN TO THE VALIDATION SAMPLE (17 SAMPLES).

	Experiment 2A	Experiment 2B
Avg training accuracy /Std:	95.8334 +/- 3.0138	97.2223 +/-1.5713
Avg test accuracy / Std:	76.4706 +/- 10.8920	78.6765 +/- 8.8585
Avg number of genes/ Std:	21.1250 +/- 6.2892	9.7500 +/- 2.2520

When the eight subsets derived from experiment 2B were combined and repeated genes were removed, the remaining subset included 37 genes. The validation set was classified with the 3-NN classifier including only the expression values from these 37 genes. The accuracy was 94.12%, which is higher than the test accuracies on any of the eight subsets that the set of 37 genes was built from.

This result seems better than those reported previously on this dataset. Although our results are not directly comparable, as noted before we can make qualified comparisons. It is very interesting that using EA/*k*-NN alone combined with some straightforward analysis we have found a subset whose validation set accuracy is better than test accuracies reported in previous work on these data.

TABLE V. A RANKED LIST OF THE GENES WHICH WERE THE MOST FREQUENTLY SELECTED IN THE FINAL SUBSETS OF EXPERIMENT 2B. THE THIRD COLUMN SHOWS THE OBTAINED TEST ACCURACY WHEN EACH OF THE GENES WAS USED SINGLY BY THE KNN CLASSIFIER TO CLASSIFY THE VALIDATION SET.

Rank	Gene	Test accuracy
1	*T72175*	52.9412 %
2	T52342	47.0588 %
3	*J00231,*	47.0588 %
4	*X60489,* T58861	64.7059 % 64.7059 %

TABLE VI. TEST ACCURACY ON VALIDATION SET WHEN USING ONLY THE MOST FREQUENTLY SELECTED GENES FROM EXPERIMENT 2B.

Genes	Test accuracy
T72175, T52342*, J00231, X60489,* T58861	76.4706 %
J00231, X60489, T58861	58.8235 %
T72175, T52342	47.0588 %

The eight subsets generated in experiment 2B had many genes in common. *T72175* was the most frequent, in all 8 sets. *T52342* was in 7 out of 8 subsets. Each of *J00231, X60489 and T58861* were in 4 of 8 subsets, and *UMGAP* was in three subsets. Additional tests were carried out on the validation set using these five genes. The genes were ranked based upon how well they could classify the samples in the validation set (Table IV). Test accuracy using each of the genes singly is also illustrated in the table.

The best test accuracy (94.12%) was obtained when all the unique genes from expt 2B were pooled to form a subset of 37 genes. The highest test accuracy obtained for two of the eight subsets was 88.23% including 9 and 13 genes respectively. The 3-NN classifier returned a test accuracy of only 76.47% when only the five frequent genes were included (see Table VI). Combined, the two top ranked genes (*T72175,* T52342) achieved only 47.05%.

D. Brief Analysis of Selected Genes

Unlike in the prostate cancer experiments, in this case the small collection of most frequently occurring genes do not in themselves constitute a particularly high-accuracy feature set (Column 1 in Table VI). We nevertheless providing brief notes of annotation for each of these five genes. It is notable that not all of these genes have been previously indicated as significant for colon cancer.

T72175: immunoglobulin kappa constant ;

J00231: immunoglobulin gamma 3 ;

T58861 is a ribosomal protein L30. Ribosomal Protein genes have been found to be over-expressed in colorectal tumors [32]. ;

T52342 : Human tra1 mRNA for human homologue of murine tumor rejection antigen gp96. ;

X60489: Human mRNA for elongation factor-1-beta

VI. Discussion/Conclusion

Many methods have been explored which combine feature selection and classification, in order to analyse and interpret highly significant data (such as microarray datasets), that are overwhelmingly blessed with features, most of which are believed to be redundant and/or insignificant in relation to the precise issue under study. It has generally been found (and is not surprising) that prior feature selection is beneficial before running either classification methods on the selected features, or running a further `combined' feature selection/classification method. In this paper we have explored the simple notion of using a combined EA/*k*-NN approach for *both* prior feature selection and a second feature selection/classification phase. By combining this strategy with further straightforward analysis of features (genes) that occur often in repeated experiments, we are able to find gene subsets whose test set accuracy (taking into account the different designs of previously reported experiments on the same data) is highly competitive. This is both useful from the viewpoint of building predictive models, as well as useful from the viewpoint of finding potentially significant genes. In particular, we argue that by eschewing a sophisticated nonlinear classification method, success using the *k*-NN classifier is likely to be more dependent on finding significant subsets. Further, owing to the generally good performance of EA/*k*-NN when used once, *without any initial dimensionality reduction*, we can have some confidence that its use purely as a feature selection technique is likely to yield a `good' reduced set of genes in this sense.

The basic performance of EA/*k*-NN in this two-phase sense appears validated since it has led directly to the identification of a 2-gene subset that has 88.89% accuracy on the validation set for prostate cancer, and on the colon cancer dataset it has yielded a 37-gene subset with 94.23% test set performance. These results are highly competitive with those in recent published work on the same datasets, taking into account the experimental setups. Finally, we have identified three genes in relation to prostate cancer, and five genes in relation to colon cancer, which respectively support previous findings and suggest putative targets for further research.

Acknowledgements

The authors are grateful to Evosolve (UK registered charity no. 1086384) for supporting part of the work reported here. The authors are also very grateful for the helpful comments of the anonymous reviewers.

References

[1] Pease, A.C, Solas, D., Sullivan, E.J.,Cronin, M.T., Holmes, C.P., Fodor, S.P.A. 1994. Light-directed oligonucleotide arrays for rapid DNA sequence analysis. *Proc. Natl. Acad. Sci. USA*. **91**: 5022-5026.

[2] Alon, U., Barkai, N., Notterman, D.A., Gish, K., Ybarra, S., Mack, D., Levine, A.J., 1999. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proc. Natl Acad. Sci. USA*. **96**: 6745–6750

[3] Golub, T.R., Slonim, D.K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J.P., Coller, H., Loh, M.L., Downing, J.R., Caligiuri,M.A., Bloomfield, C.D., Lander, E.S., 1999. Molecular classi-fication of cancer: Class discovery and class prediction by gene expression monitoring. *Science.* **286**: 531–537.

[4] Perou CM, Sorlie T, Eisen MB, van de Rijn M, Jeffrey SS, Rees CA, Pollack JR, Ross DT, Johnsen H, Akslen LA, Fluge O, Pergamenschikov A, Williams C, Zhu SX, Lonning PE, Borresen-Dale AL, Brown PO, Botstein D. 2000. Molecular portraits of human breast tumours. *Nature*, **406:**747-752.

[5] Alizadeh, A.A., Eisen, M. B., Davis, R. E., Ma, C. *et. al.* 2000. Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling, *Nature*, vol. **403**, 503-511.

[6] Li. L, Darden, A.D, Weinberg, C. R., Levine, A. J., Pedersen, L.G. (2001) Gene Assessment and Sample Classification for Gene Expression Data Using a Genetic Algorithm/k-nearest Neighbor Method. *Combinatorial Chemistry & High Throughput Screening.* **4**, 727-739.

[7] Siedlecki, W., Sklansky, J. 1989. A note on genetic algorithms for large-scale feature selection. *Pattern Recognition Letters.* **10**: 335-347.

[8] Kelly, J. D and Davis, L. 1991. Hybridizing the Genetic Algorithm and the K Nearest Neighbors Classification Algorithm. *Proc. Fourth Inter. Conf. Genetic Algorithms and their Applications (ICGA)*, 377-383.

[9] Punch, W.F., Pei, M., Chia-Shun, L., Goodman, E.D., Hovland, P., and Enbody R. 1993. Further research on Feature Selection and Classification Using Genetic Algorithms. *In 5th International Conference on Genetic Algorithm, Champaign I.,* p.557-564

[10] Guyon, I., Weston, J., Barnhill, S., Vapnik, V. 2002 **Gene** selection for cancer classification using support vector machines. *Machine Learning*, **46:**389-422.

[11] Liu, H., Yu, L. 2005. Toward Integrating Feature Selection Algorithms for Classification and Clustering. *IEEE Transactions on Knowledge and Data Engineering*, **17(4):** 491-5020.

[12] Raymer, M.L., Punch, W.F., Goodman, E.D., Kuhn, L.A. 1996. Genetic Programming for Improved Data Mining – Application to the Biochemistry of Protein Interactions. *in Genetic Programming 1996: Proceedings* MIT Press, Cambridge, MA, p.275–381.

[13] Dudoit, S., Fridlyand, J., Speed, T. 2000. Comparison of discrimination methods for the classification of tumors using gene expression data, Technical Report, Berkeley.

[14] Wang Y, Makedon FS, Ford JC, Pearlman J. 2005. HykGene: a hybrid approach for selecting marker genes for phenotype classification using microarray gene expression data. Bioinformatics. **21**(8):1530-1537.

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

[16] Fogel, L.J., A.J. Owens, and M.J. Walsh (1966) *Artificial Intelligence Through Simulated Evolution*, John Wiley, New York.

[17] Schwefel, H.-P. (1981) *Numerical Optimization of Computer Models*, John Wiley, Chichester, U.K.

[18] Holland, J. H. (1975) *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, MI.

[19] Goldberg, D. 1989. Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Weslay, New York.

[20] Fix, E., Hodges, J. 1951. Discriminatory analysis: Nonparametric discrimination: Consistency properties. Technical report 21-49-004, USAF School of Aviation Medicine.

[21] Raymer, M.L., Punch, W.F., Goodman, E.D., Kuhn, L.A, Jain, A. K. 2000. Dimensionality Reduction Using Genetic Algorithms. *IEEE Transactions on Evolutionary Computation*, **4**: 164–171.

[22] Lu, Y., Han, J. 2003. Cancer classification using gene expression data. *Inform. Syst.,* **28**, 243-268.

[23] Li. L, Weinberg, C. R., Darden, A.D, Pedersen, L.G. 2001. Gene selection for sample classification based on gene expression data: study of sensitivity to choice of parameters of the GA/KNN method. *Bioinformatics*. **17**, 1131-1142.

[24] Jirapech-Umpai, T., Aitken, S. 2005. Feature selection and classification for microarray data analysis: Evolutionary methods for identifying predictive genes. *BMC Bionformatics*. 6:148.

[25] Singh, D., Febbo, P. G., Ross, K., Jackson, D. G., Manola, J, Ladd, C., Tamayo, P, Renshaw, A. A., D'Amico, A. V., Richie, J. P. Lander, E. S, Loda, M. Kantoff, P.W, Golub, T. R, Sellers, W.R. 2002. Gene expression correlates of clinical prostate cancer behavior. *Cancer Cell*, **1**: 203-209

[26] Topon, K. P. Iba, H. 2005. Extraction of Informative Genes from microarray data. Available online: http://www.iba.k.u-tokyo.ac.jp/english/papers/2005/toponGECCO2005.pdf .

[27] Liu, B., Cui, Q., Jiang, T. , Ma, S. 2004. A combinational feature selection and ensemble neural network method for classification of gene expression data. *BMC Bioinformatics.* **5**:136

[28] Dettling M, Buhlmann P: Boosting for tumor classification with gene expression data. *Bioinformatics* 2003, 19:1061-1069.

[29] Holzbeierlein, J. Lal, P. Tulippe, E. L., Smith, A., Satagopan, J. Zhang, L. Ryan,C., Smith, S. Scher, H., Scardino,P. Reuter, V., Gerald, W. L. 2004. Gene Expression Analysis of Human Prostate Carcinoma during Hormonal Therapy Identifies Androgen-Responsive Genes and Mechanisms of Therapy Resistance. *American Journal of Pathology,* **164:** (1)

[30] Liu, J-W., Shen J-J., Tanzillo-Swarts, A, Bhatia, B., Maldonado, C., Person, M.D., Lau, S., and Tang, D.G. 2003. Annexin II expression is reduced or lost in prostate cancer cells and its re-expression inhibits prostate cancer cell migration. *Oncogene* **22:** 1475-1485.

[31] Epner, D. E., Coffey, D. S. 1996. There are multiple forms of glyceraldehydes-3-phosphate dehydrogenase in prostate cancer cells and normal prostate tissue. *Prostate.* **28** (6): 372-378.

[32] Pogue-Geile, K.; Geiser, J. R.; Shu, M.; Miller, C.; Wool, I. G.; Meisler, A. I.; Pipas, J. M. (1991)Ribosomal protein genes are overexpressed in colorectal cancer: isolation of a cDNA clone encoding the human S3 ribosomal protein. Molec. Cell. Biol. 11: 3842-3849, 1991.