REVIEW

Ensemble machine learning on gene expression data for cancer classification

Aik Choon Tan and David Gilbert

Bioinformatics Research Centre, Department of Computing Science, University of Glasgow, Glasgow, UK

Abstract: Whole genome RNA expression studies permit systematic approaches to understanding the correlation between gene expression profiles to disease states or different developmental stages of a cell. Microarray analysis provides quantitative information about the complete transcription profile of cells that facilitate drug and therapeutics development, disease diagnosis, and understanding in the basic cell biology. One of the challenges in microarray analysis, especially in cancerous gene expression profiles, is to identify genes or groups of genes that are highly expressed in tumour cells but not in normal cells and vice versa. Previously, we have shown that ensemble machine learning consistently performs well in classifying biological data. In this paper, we focus on three different supervised machine learning techniques in cancer classification, namely C4.5 decision tree, and bagged and boosted decision trees. We have performed classification tasks on seven publicly available cancerous microarray data and compared the classification/prediction performance of these methods. We have observed that ensemble learning (bagged and boosted decision trees) often performs better than single decision trees in this classification task.

Keywords: supervised machine learning, ensemble methods, cancer classification, gene expression data, performance evaluation

Introduction

Recent technological advancement in molecular biology – especially microarray analysis – on whole genome RNA expression facilitates new discoveries in basic biology, pharmacology and medicine. The gene expression profile of a cell determines its phenotype, function and response to the environment. Quantitative measurement of gene expression can potentially provide clues about the mechanisms of gene regulation and interaction, and at the abstract level about biochemical pathways and the cellular function of a cell. Furthermore, comparison between genes expressed in diseased tissue and the normal counterpart will further our understanding in the disease pathology, and also help to identify genes or groups of genes as targets for potential therapeutic intervention.

Although mRNA is not the ultimate product of a gene, transcription is the first step in gene regulation, and this information is important for understanding gene regulatory networks. Obtaining measurements of mRNA is considerably cheaper, and can be more easily carried out in a high-throughput manner, compared to direct measurements of the protein levels. There may not be strong evidence to suggest a correlation between the mRNA and abundance of proteins in a cell; but absence of mRNA in a cell is likely to imply a low level of the respective protein. Thus, the qualitative estimation of a proteome can be based on the quantitative measurement of the transcriptome (Brazma and Vilo 2000).

Gene expression data can be obtained by high-throughput technologies such as microarray and oligonucleotide chips under various experimental conditions, at different developmental stages or in different tissues. The data are usually organised in a matrix of n rows and m columns, which is known as a gene expression profile. The rows represent genes (usually genes of the whole genome), and the columns represent the samples (eg various tissues, developmental stages and treatments). One can carry out two straightforward studies by comparing the genes (n rows) or comparing the samples (m columns) of the matrix (Figure 1). If we find that two rows are similar, we can hypothesise that the two genes are co-regulated and possibly functionally related. These analyses may facilitate our understanding of gene regulation, metabolic and signalling pathways, the genetic mechanisms of disease, and the response to drug treatments.

Considering the amount and complexity of the gene expression data, it is impossible for an expert to compute and compare the $n \times m$ gene expression matrix manually

Correspondence: Aik Choon Tan, Bioinformatics Research Centre, Department of Computing Science, University of Glasgow, 17 Lilybank Gardens, Glasgow G12 8QQ, UK; tel +44 141 330 2421; fax +44 141 330 3690; email actan@brc.dcs.gla.ac.uk

m samples

Geneid	Condition 1	Condition 2	...	Condition *m*
Gene1	103.02	58.79	...	101.54
Gene2	40.55	1246.87	...	1432.12
...	...	...	...	...
Gene *n*	78.13	66.25	...	823.09

n genes

Figure 1 A typical gene expression matrix where rows represent genes (usually genes of the whole genome), and the columns represent the samples (eg various tissues, developmental stages and treatments).

(where n is usually greater than 5000 and m is more than 10). Thus, machine learning and other artificial intelligence techniques have been widely used to classify or characterise gene expression data (Golub et al 1999; Ben-dor et al 2000; Brazma and Vilo 2000; Brown et al 2000; Li and Wong 2002; Shipp et al 2002). This is due to the nature of machine learning approaches in that they perform well in domains where there is a large amount of data but little theory – this is exactly the situation in analysing gene expression profiles.

Machine learning is the sub-field of artificial intelligence which focuses on methods to construct computer programs that learn from experience with respect to some class of tasks and a performance measure (Mitchell 1997). Machine learning methods are suitable for molecular biology data due to the learning algorithm's ability to construct classifiers/hypotheses that can explain complex relationships in the data. Generally, there are two types of learning schemes in machine learning: supervised learning, where the output has been given and labelled a priori or the learner has some prior knowledge of the data; and unsupervised learning, where no prior information is given to the learner regarding the data or the output. Ensemble machine learning is a method that combines individual classifiers in some way to classify new instances.

The objective of this study is to investigate the performance of ensemble machine learning in classifying gene expression data on cancer classification problems. This paper outlines the materials and methods used in this study, presents the results and discusses the observation from the results. The final section summarises this study.

Materials and methods

The challenge of the treatment of cancer has been to target specific therapies to pathogenetically distinct tumour types, in order to maximise efficacy and minimise toxicity (Golub et al 1999). Cancer classification has been the central topic of research in cancer treatment. The conventional approach for cancer classification is primarily based on the morphological appearance of the tumour. The limitations for this approach are the strong bias in identifying the tumour by experts and also the difficulties in differentiating between cancer subtypes. This is due to most cancers being highly related to the specific biological insights such as responses to different clinical treatments. It therefore makes biological sense to perform cancer classification at the genotype level compared to the phenotypic observation. Due to the large amount of gene expression data available on various cancerous samples, it is important to construct classifiers that have high predictive accuracy in classifying cancerous samples based on their gene expression profiles. Besides being accurate, the classifier needs to provide explanations to the biologists/clinicians about the relationship between the selected discriminative genes. We have employed decision trees in this study due to the easy interpretation of the final classifiers, compared to other 'black-box' approaches (eg artificial neural networks).

Basic notations

The training examples for supervised machine learning are in the form of a set of tuples $<x, y>$ where y is the class label and x is the set of attributes for the instances. The attributes for cancerous classification will be the gene expression signals, and the class consisting of cancerous or normal tissues. The learning algorithm is trained on the positive E^+ (cancerous samples) and negative E^- (normal samples) examples to construct a classifier $C(x)$ that distinguishes between these examples. In the ideal case, $E^+ \cap E^- = \varnothing$.

Problem formulation

The objective of this study is to construct classifiers that can correctly classify the cancerous tissues and normal tissues from the gene expression profiles. The learner needs to construct a classifier to distinguish between the cancerous samples and the normal samples. This classifier can then be used as the basis for classifying as yet unseen clinical samples in the future. This is a classical supervised learning problem that applies a learning algorithm on the training data and performs prediction on the test data. In this study, we only consider supervised machine learning applied to cancer classification.

Machine learning algorithms

We have applied single C4.5 (Quinlan 1993), Bagging and AdaBoost decision trees to classify seven publicly available gene expression datasets. All the learning methods used in this study were obtained from the WEKA machine learning package (Witten and Frank 2000) (http://www.cs.waikato.ac.nz/~ml/weka/).

C4.5 algorithm

The decision tree algorithm is well known for its robustness and learning efficiency with its learning time complexity of $O(n \log_2 n)$. The output of the algorithm is a decision tree, which can be easily represented as a set of symbolic rules (IF … THEN …). The symbolic rules can be directly interpreted and compared with existing biological knowledge, providing useful information for the biologists and clinicians.

The learning algorithm applies a divide-and-conquer strategy (Quinlan 1993) to construct the tree. The sets of instances are accompanied by a set of properties (attributes). A decision tree is a tree where each node is a test on the values of an attribute, and the leaves represent the class of an instance that satisfies the tests. The tree will return a 'yes' or 'no' decision when the sets of instances are tested on it. Rules can be derived from the tree by following a path from the root to a leaf and using the nodes along the path as preconditions for the rule, to predict the class at the leaf. The rules can be pruned to remove unnecessary preconditions and duplication. Figure 2 shows a decision tree induced from colon tumour data and also the equivalent decision rules.

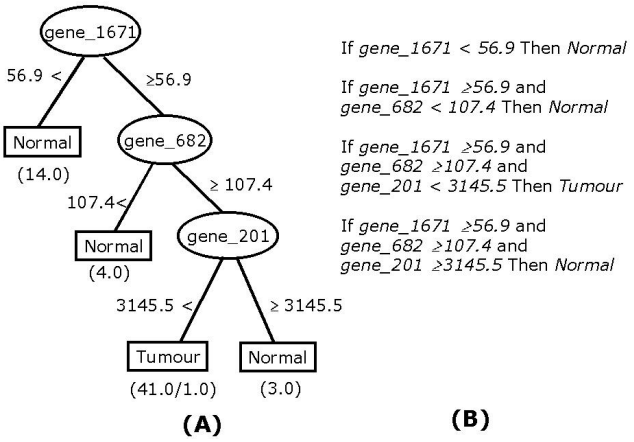

Figure 2 (A) A decision tree induced from the colon tumour data set. The nodes represent genes, and branches represent the expression conditions. The leaves of the tree represent the decision outcome (in this case either 'is a tumour tissue' or 'is a normal tissue'). The brace under a leaf denotes the number of instances correctly and incorrectly classified by the leaf (TP/FP). **(B)** The equivalent decision rules are derived from the decision trees.

Ensemble methods

We regard ensemble methods as sets of machine learning techniques whose decisions are combined in some way to improve the performance of the overall system. Other terminologies found in the literature to denote similar meanings are: multiple classifiers, multi-strategy learning, committee, classifier fusion, combination, aggregation, integration and so on. In this paper, we use ensemble to refer to all the classifier combination methods. The simplest way to combine different learning algorithms is by voting or weighted voting.

The intuitive concept of ensemble learning is that no single approach or system can claim to be uniformly superior to any other, and that the integration of several single approaches will enhance the performance of the final classifier (eg accuracy, reliability, comprehensibility). Hence, an ensemble classifier can have overall better performance than the individual base classifiers. The effectiveness of ensemble methods is highly reliant on the independence of the error committed by the individual base learner. The performance of ensemble methods strongly depends on the accuracy and the diversity of the base learners. Various studies (Breimen 1996; Bauer and Kohavi 1999; Dietterich 2000a, 2000b) have shown that decision trees tend to generate diverse classifiers with response to small changes in the training data and, are therefore, suitable candidates for the base learner of an ensemble system. The easiest approach to generate diverse base classifiers is manipulating the training data. In this study, we investigate bagging and boosting, the two most common ensemble techniques.

Bagging (**b**oostrap **agg**regat**ing**) was introduced by Breimen (1996) and it aims to manipulate the training data by randomly replacing the original T training data by N items. The replacement training sets are known as bootstrap replicates in which some instances may not appear while others appear more than once. The final classifier $C^*(x)$ is constructed by aggregating $C_i(x)$ where every $C_i(x)$ has an equal vote. The bagging algorithm is shown in Figure 3.

Freund and Schapire (1996) introduced AdaBoost (**Ada**ptive **Boost**ing) method as an alternative method to influence the training data. Initially, the algorithm assigns every instance x_i with an equal weight. In each iteration i, the learning algorithm tries to minimise the weighted error on the training set and returns a classifier $C_i(x)$. The weighted

Input: Training examples <x, y>, Machine Learning Algorithm *ML*, Integer j (number of iteration)

1. For each iteration $i = 1 \ldots j$
2. {
3. Select a subset t of size N from the original training examples T
4. The size of t is the same with the T where some instances may not appear in it while others appear more than once (re-sampling)
5. Generates a classifier $C_i(x)$ from the t
6. }
7. The final classifier $C^*(x)$ is formed by aggregating the j classifiers
8. To classify an instance x, a vote for class y is recorded by every classifier $C_i(x) = y$
9. $C^*(x)$ is the class with the most votes. (Ties being resolved arbitrarily.)

Output: $C^*(x)$

Figure 3 Bagging algorithm.

error of $C_i(x)$ is computed and applied to update the weights on the training instances x_i. The weight of x_i increases according to its influences on the classifier's performance that assigns a high weight for a misclassified x_i and a low weight for a correctly classified x_i. The final classifier $C^*(x)$ is constructed by a weighted vote of the individual $C_i(x)$ according to its accuracy based on the weighted training set. Figure 4 illustrates the AdaBoost algorithm.

Dataset

In this section, we briefly describe the gene expression datasets used in this study. Interested readers should refer to the references for the details of the microarray experiment setup and methodologies in acquiring the expression data. These datasets were obtained from the Gene Expression Datasets Collection (http://sdmc.lit.org.sg/GEDatasets/). The gene expression datasets are summarised in Table 1.

Input: Training examples <x, y>, Machine Learning Algorithm *ML*, Integer j (number of iteration)

1. Assigns an equal weight for instance x_i
2. For each iteration $i = 1 \ldots j$
3. {
4. Generates a classifier $C_i(x)$ with minimise the weighted error over the instances x
5. Update the weight of x_i
6. }
7. The final classifier $C^*(x)$ is formed by a weighted vote of the individual $C_i(x)$ according to its accuracy on the weighted training set

Output: $C^*(x)$

Figure 4 AdaBoost algorithm.

1. *Acute lymphoblastic leukaemia (ALL)* and *acute myeloid leukaemia (AML)* (Golub et al 1999). The learning objective of this gene expression data is to perform cancer subtype classification. The data consists of two distinctive acute leukaemias, namely AML and ALL bone marrow samples. There are over 7129 probes from 6817 human genes for this experiment. The training dataset consists of 38 samples (27 ALL and 11 AML), and the test data consists of 34 samples (20 ALL and 14 AML).

2. *Breast cancer outcome* (van't Veer et al 2002). The objective of learning over this gene expression dataset is to predict the patient clinical outcome after their initial diagnosis for an interval of at least 5 years. The training data contains 78 patient samples, 34 of which are from patients who had developed distance metastases within 5 years (relapse) and 44 samples are from patients who remained healthy after the initial diagnosis (non-relapse). The test data consists of 12 relapse and 7 non-relapse samples. The numbers of genes used in this study was 24 481.

3. *Central nervous system (CNS) embryonal tumour outcome* (Pomeroy et al 2002). The experimental objective of this dataset is to classify the patients who are alive after treatment ('survivors') and those who succumbed to their disease ('failures'). This gene expression data is dataset C in the paper, which consists of 60 patient samples (21 survivors and 39 failures). There are 7129 probes from 6817 human genes in the dataset.

4. *Colon tumour* (Alon et al 1999). The aim of this gene expression experiment is to construct a classifier that can classify colon tumour from normal colon tissues. The dataset consists of 40 colon tumour tissues and 22 normal tissues. There are 7129 probes from 6817 human genes in the dataset.

5. *Lung cancer* (Gordon et al 2002). This set of gene expression data consists of the lung malignant pleural mesothelioma (MPM) and adenocarcinoma (ADCA) samples. The learning objective of this dataset is to construct a classifier that can distinguish between these two tumour classes. The training data consists of 32 samples (16 MPM and 16 ADCA) while the testing data consists of 149 samples (15 MPM and 134 ADCA). There are 12 533 probes in this dataset.

6. *Prostate cancer* (Singh et al 2002). The classification task of this dataset is to construct a classifier that can predict a prostate tumour from gene expression data. The training data consists of 52 prostate tumour tissues and 50 normal

Table 1 Summary of the gene expression data. The positive and negative examples column represents the number of positive and negative examples in training set, test set and the name of the positive and negative class, respectively

Dataset	*Continuous attributes (nr of genes)*	*Positive examples (train:test:class)*	*Negative examples (train:test:class)*
ALL/AML leukaemia	7129	27:20:ALL	11:14:AML
Breast cancer outcome	24 481	34:12:relapse	44:7:non-relapse
CNS embryonal tumour outcome	7129	21:0:survivors	39:0:failures
Colon tumour	7129	40:0:tumour	22:0:normal
Lung cancer	12 533	16:15:MPM	16:134:ADCA
Prostate cancer	12 600	52:25:tumour	50:9:normal
Prostate cancer outcome	12 600	8:0:relapse	13:0:non-relapse

tissues. The testing data were obtained from a different experiment (Welsh et al 2001) which consists of 25 tumour samples and 9 normal samples. There are 12 600 genes and expressed sequence tags (ESTs) in this dataset.

7. *Prostate cancer outcome* (Singh et al 2002). This dataset consists of gene expression data from patients with respect to recurrence following surgery. The training data contains 8 patients having relapsed and 13 patients having remained relapse free ('non-relapse') for at least 4 years. There are 12 600 genes and ESTs in this dataset.

Methods

Step 1 Filtering: discretization of continuous-valued attributes

Gene expression data contains a lot of 'noise' or irrelevant signals, and so one of the important steps in machine learning is to perform data cleaning or feature extraction before the actual learning process. We employed Fayyad and Irani's (1993) discretization method to filter out the noise. This algorithm recursively applies an entropy minimisation heuristic to discretize the continuous-valued attributes. The stopping criterion for this algorithm is based on the minimum description length principle (MDL). This method was found to be quite promising as a global discretization method (Ting 1994), as Li and Wong (2002) employed this technique to filter out the non-discriminatory genes before performing classification on the gene expression data. After the filtering process, we observed that the data size reduces to 50%–98% of the actual data. This indicates that most of the genes play an irrelevant part in cancer classification problems. Table 2 shows the size of the data before and after filtering, and also the percentage of genes that are used in the actual learning process.

Table 2 Summary of the filtered data

Dataset	*Before filtering (nr of genes) continuous*	*After filtering (nr of genes) discrete*	*Percentage of important genes (%)*
ALL/AML leukaemia	7129	1038	14.56
Breast cancer outcome	24 481	834	3.41
CNS embryonal tumour outcome	7129	74	1.04
Colon tumour	7129	135	1.89
Lung cancer	12 533	5365	42.81
Prostate cancer	12 600	3071	24.37
Prostate cancer outcome	12 600	208	1.65

Step 2 Classification/prediction of positive and negative examples

After the filtering process, we employed three different learning algorithms to construct the classifier, namely single C4.5, bagged C4.5 ('bagging') and AdaBoost C4.5 ('boosting'). For the datasets of leukaemia, breast cancer outcome, lung cancer and prostate cancer the learning algorithm performs prediction on the test data. For the CNS embryonal tumour outcome, colon tumour and prostate cancer outcome where no test data are available, tenfold cross-validation was performed on the training data to obtain a statistically reliable predictive measurement.

The normal method to evaluate the robustness of the classifier is to perform cross-validation on the classifier. Tenfold cross-validation has been proved to be statistically good enough in evaluating the performance of the classifier (Witten and Frank 2000). In tenfold cross-validation, the training set is equally divided into 10 different subsets. Nine out of ten of the training subsets are used to train the learner, and the tenth subset is used as the test set. The procedure is repeated ten times, with a different subset being used as the test set.

Step 3 Evaluation

The predictive accuracy of the classifier measures the proportion of correctly classified instances:

$$Acc = \frac{TP + TN}{TP + TN + FP + FN}$$

where true positives (TP) denote the correct classifications of positive examples; true negatives (TN) are the correct classifications of negative examples; false positives (FP) represent the incorrect classification of negative examples

into the positive class; and false negatives (FN) are the positive examples incorrectly classified into the negative class. Positive predictive accuracy (PPV), or the reliability of positive predictions of the induced classifier, is computed by:

$$PPV = \frac{TP}{TP + FP}$$

Sensitivity (S_n) measures the fraction of actual positive examples that are correctly classified:

$$S_n = \frac{TP}{TP + FN}$$

while specificity (S_p) measures the fraction of actual negative examples that are correctly classified:

$$S_p = \frac{TN}{TN + FP}$$

Results

Table 3 summarises the predictive accuracy of the classification methods on all the data; the highlighted values represent the highest accuracy obtained by the method. Solely based on the predictive accuracy (Table 3) of these methods, we observed that bagging constantly performs better than both boosting and single C4.5. Out of the 7 classification/prediction problems, bagging wins in 3 datasets (lung cancer, prostate cancer and prostate cancer outcome) and ties with boosting in 2 datasets (breast cancer outcome and CNS embryonal tumour outcome). Single C4.5 wins overall the ensemble methods in the colon dataset and ties with the ensemble methods in the ALL/AML leukaemia dataset.

Table 3 Predictive accuracy of the classifiers[a]

	Predictive accuracy (%)		
Dataset	*Single C4.5*	*Bagging C4.5*	*AdaBoost C4.5*
ALL/AML leukaemia	**91.18**[a]	**91.18**[a]	**91.18**[a]
Breast cancer outcome	63.16	**89.47**[a]	**89.47**[a]
CNS embryonal tumour outcome	85.00	**88.33**[a]	**88.33**[a]
Colon tumour	**95.16**[a]	93.55	90.32
Lung cancer	92.62	**93.29**[a]	92.62
Prostate cancer	67.65	**73.53**[a]	67.65
Prostate cancer outcome	52.38	**85.71**[a]	76.19

[a] Denotes method that has the highest accuracy.

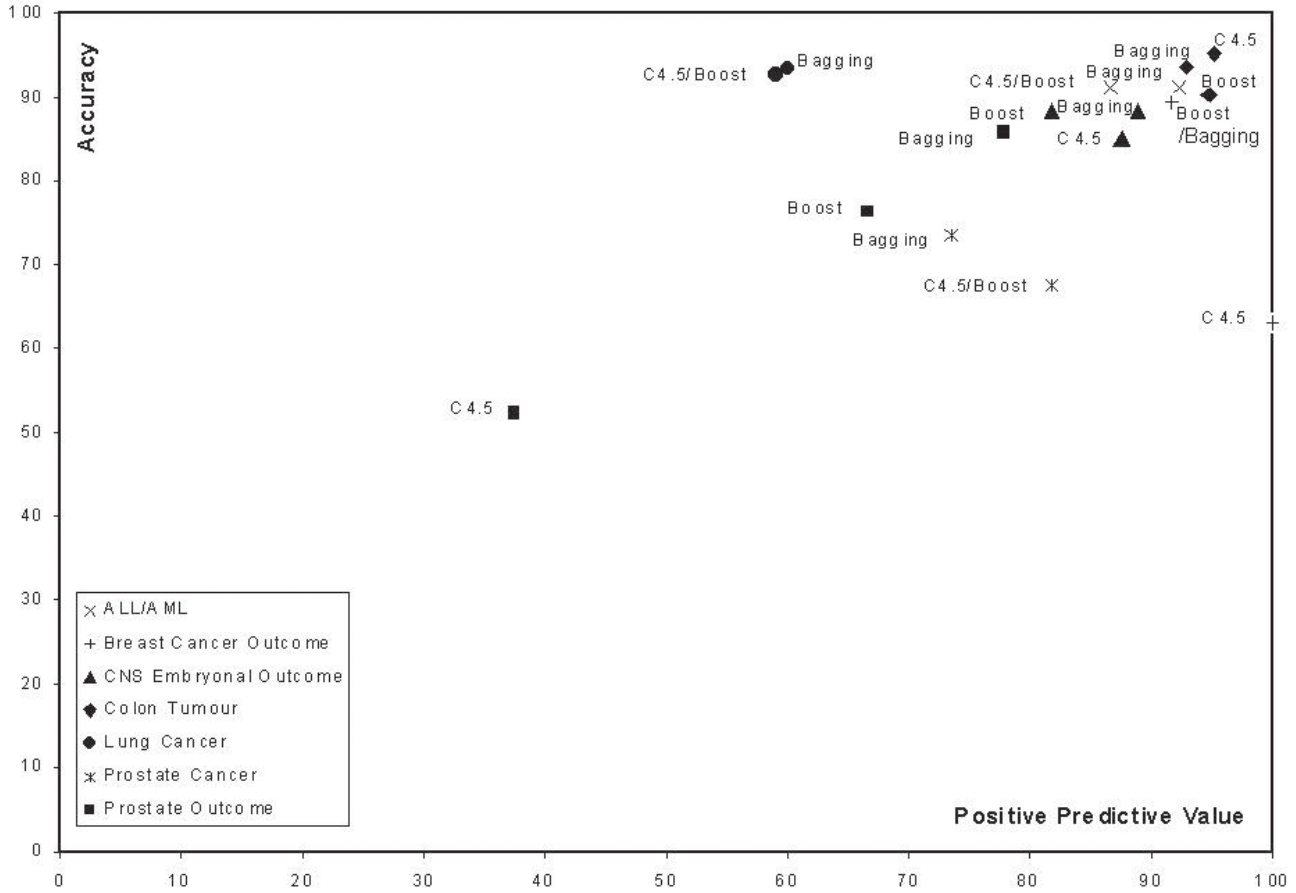

Figure 5 Comparison of the single C4.5, Bagging C4.5 (bagging) and AdaBoost C4.5 (boosting) Predictive accuracy (Acc) and Positive predictive value (PPV) on the seven cancerous gene expression data.

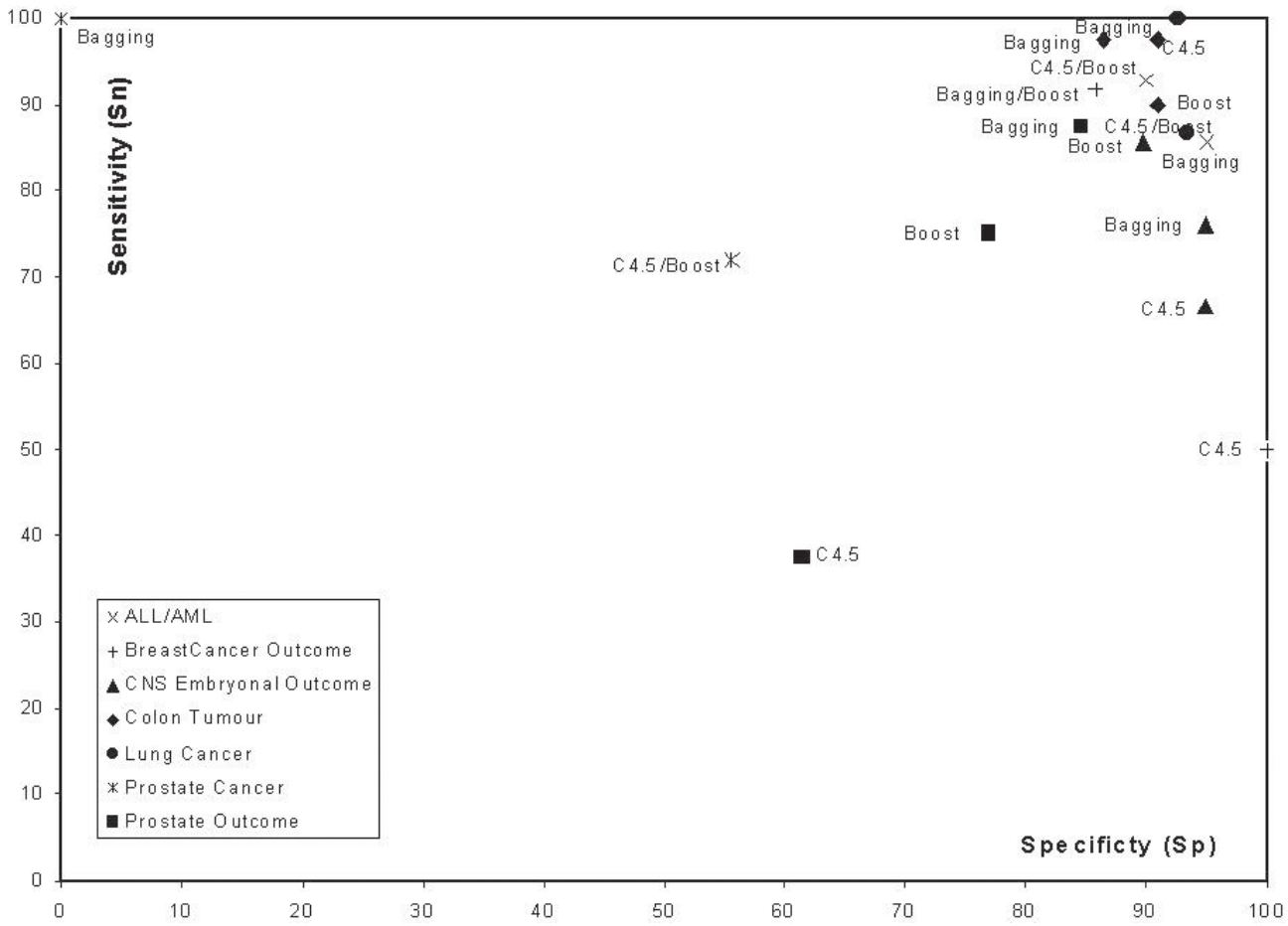

Figure 6 Comparison of the single C4.5, Bagging C4.5 (bagging) and AdaBoost C4.5 (boosting) Sensitivity (S_n) and Specificity (S_p) on the seven cancerous gene expression data.

Figure 5 illustrates the predictive accuracy (Acc) versus positive predictive value (PPV) of the different methods. Figure 6 shows the sensitivity (S_n) and specificity (S_p) of the methods in classifying the gene expression data. From Figure 6, we observe that bagging method applied to prostate cancer obtained 100% sensitivity but 0% specificity. This is due to the fact that the bagging classifier predicts all the test data as tumour tissues. In this specific case, the classifier is unable to distinguish between the tumour tissues and the normal tissues. However, considering the other two methods (single C4.5 and AdaBoost C4.5), both obtain the same predictive accuracy and at the same time increase the specificity to 56% while retain a relatively good sensitivity (72%). This shows that the latter methods (single C4.5 and AdaBoost C4.5) perform much better compared to bagging, and are capable of distinguishing the tumour tissues from the normal tissues. This observation suggests that when comparing the performance of different classifiers, one needs to take into account the sensitivity and specificity of a classifier, rather than just concentrating solely on its predictive accuracy.

Discussion

The key observation from this experiment is that none of the individual methods can claim that they are superior to the others. This is due to the algorithms' inherited statistical, computational and representational limitations (Dietterich 2000a). The learning objective of these methods is to construct a discriminative classifier, which can be viewed as finding (or approximating) the true hypothesis from the entire possible hypothesis space. We will refer to a true hypothesis as a discriminatory classifier in the rest of this paper. Every learning algorithm employs a different search strategy to identify the true hypothesis. If the size of the training example is too small (which is the case when classifying microarray data), the individual learner can induce different hypotheses with similar performances from the search space. Thus, by averaging the different hypotheses, the combined classifier may produce a good approximation to the true hypotheses (the discriminative classifier). The computational reason is to avoid local optima of the individual search strategies. The final classifier may

provide a better approximation to the true hypotheses by performing different initial searches and combining the outputs. Lastly, due to the limited amount of training data, an individual classifier may not represent the true hypothesis. Thus, through considering diverse base classifiers, it may be possible for the final classifier to approximate representation of the true hypotheses.

One interesting observation arising from this experiment is the poor performance of AdaBoost C4.5. Although data cleaning has been performed in this study, we believe that a lot of noise still remains in the training data. This is due to AdaBoost C4.5 trying to construct new decision trees to eliminate the noise (or misclassified instances) in every iteration. The AdaBoost C4.5 algorithm attempts to optimise the representational problem for the classification task. Such direct optimisation may lead to the risk of overfitting because the hypothesis space for the ensemble is much larger than the original algorithm (Dietterich 2000a). This is why AdaBoost C4.5 performs poorly in this experiment. Similar results were also observed by Dudoit et al (2002), and Long and Vega (2003).

Conversely, bagging is shown to work very well in the presence of noise. This is because the re-sampling process captures all of the possible hypotheses, and bagging tends to be biased hypotheses that give good accuracy on the training data. By averaging the hypotheses from individual classifiers, bagging increases the statistical optimisation to the true hypothesis. Thus, in this experiment, bagging consistently works well in classifying the cancerous samples.

This study shows that the true hypothesis ('discriminatory genes') of colon cancer is captured and represented by a single decision tree, thus single C4.5 outperforms the ensemble methods (Figure 2).

Various empirical observations and studies have shown that it is unusual for single learning algorithms to outperform other learning methods in all problem domains. We carried out empirical studies comparing the performance of 7 different single learners to 5 ensemble methods in classifying biological data, and showed that most of the ensemble methods perform better than an individual learner (Tan and Gilbert 2003). Our observations agree with previous ensemble studies in other domains where combined methods improve the overall performance of the individual learning algorithm.

Ensemble machine learning has been an active research topic in machine learning but is still relatively new to the bioinformatics community. Most of the machine learning-oriented bioinformatics literature still largely concentrates on single learning approaches. We believe that ensemble learning is suitable for bioinformatics applications due to the fact that the classifiers are induced from incomplete and noisy biological data. This is specifically the case when classifying microarray data where the number of examples (biological samples) is relatively small compared to the number of attributes (genes). On the other hand it is a hard-problem for the single learning method to capture true hypotheses from this type of data. Ensemble machine learning provides another approach to capture the true hypothesis by combining individual classifiers. These methods have been well tested on artificial and real data and have proved to outperform individual approaches (Breiman 1996; Freund and Schapire 1996; Quinlan 1996; Bauer and Kohavi 1999; Dietterich 2000b; Dudoit et al 2002; Long and Vega 2003; Tan and Gilbert 2003). Although these methods have been supported by theoretical studies, the only drawback for this approach is the complexity of the final classifier.

Conclusions

Machine learning has increasingly gained attention in bioinformatics research. Cancer classification based on gene expression data remains a challenging task in identifying potential points for therapeutics intervention, understanding tumour behaviour and also facilitating drug development. This paper reviews ensemble methods (bagging and boosting) and discusses why these approaches can often perform better than a single classifier in general, and specifically in classifying gene expression data. We have performed a comparison of single supervised machine learning and ensemble methods in classifying cancerous gene expression data. Ensemble methods consistently perform well over all the datasets in terms of their specificity, sensitivity, positive predicted value and predictive accuracy; and bagging tends to outperform boosting in this study. We have also demonstrated the usefulness of employing ensemble methods in classifying microarray data, and presented some theoretical explanations on the performance of ensemble methods. As a result, we suggest that ensemble machine learning should be considered for the task of classifying gene expression data for cancerous samples.

Acknowledgements

We would like to thank colleagues in the Bioinformatics Research Centre for constructive discussions, and specifically Gilleain Torrance and Yves Deville for their useful comments and proofreading of this manuscript. The University of Glasgow funded AC Tan's studentship as part of its initiative in setting up the Bioinformatics Research Centre at Glasgow.

References

Alon U, Barkai N, Notterman DA, Gish K, Ybarra S, Mach D, Levine AJ. 1999. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proc Natl Acad Sci*, 96:6745–50.

Bauer E, Kohavi R. 1999. An empirical comparison of voting classification algorithms: bagging, boosting, and variants. *Machine Learning*, 36:105–42.

Ben-Dor A, Bruhn L, Friedman N, Nachman I, Schummer M, Yakhini Z. 2000. Tissue classification with gene expression profiles. *J Comput Biol*, 7:559–83.

Brazma A, Vilo J. 2000. Gene expression data analysis. *FEBS Lett*, 480:17–24.

Breiman L. 1996. Bagging predictors. *Machine Learning*, 24:123–40.

Brown MPS, Grundy WN, Lin D, Cristianini N, Sugnet CW, Furey TS, Ares M Jr, Haussler D. 2000. Knowledge-based analysis of microarray gene expression data by using support vector machines. *Proc Natl Acad Sci*, 97:262–7.

Dietterich TG. 2000a. Ensemble methods in machine learning. In Proceedings of the First International Workshop on Multiple Classifier Systems, MCS. 2000 Jun; Cagliari, Italy. *LNCS*, 1857:1–15.

Dietterich TG. 2000b. An experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting, and randomization. *Machine Learning*, 40:139–57.

Dudoit SJ, Fridlyand J, Speed T. 2002. Comparison of discrimination methods for the classification of tumors using gene expression data. *J Am Stat Assoc*, 97:77–87.

Fayyad UM, Irani KB. 1993. Multi-interval discretization of continuous-valued attributes for classification learning. In Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence. 1993 Aug 28–Sept 3; Chambery, France. San Francisco: Morgan Kaufmann. p 1022–7.

Freund Y, Schapire RE. 1996. Experiments with a new boosting algorithm. In Proceedings of the Thirteenth International Conference on Machine Learning. 1996 Jul 3–6; Bari, Italy. San Francisco: Morgan Kaufmann. p 148–56.

Golub TR, Slonim DK, Tamayo P, Huard C, Gaasenbeek M, Mesirov JP, Coller H, Loh ML, Downing JR, Caligiuri MA et al. 1999. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, 286:531–7.

Gordon GJ, Jensen RV, Hsiao L-L, Gullans SR, Blumenstock JE, Ramaswamy S, Richard WG, Sugarbaker DJ, Bueno R. 2002. Translation of microarray data into clinically relevant cancer diagnostic tests using gene expression ratios in lung cancer and mesothelioma. *Cancer Res*, 62:4963–7.

Li J, Wong L. 2002. Identifying good diagnostic gene groups from gene expression profiles using the concept of emerging patterns. *Bioinformatics*, 18:725–34.

Long PM, Vega VB. 2003. Boosting and microarray data. *Machine Learning*, 52:31–44.

Mitchell T. 1997. Machine learning. New York: McGraw-Hill.

Quinlan JR. 1993. C4.5: programs for machine learning. San Francisco: Morgan Kaufmann.

Quinlan JR. 1996. Bagging, boosting, and C4.5. In Proceedings of the Thirteenth National Conference on Artificial Intelligence (AAAI 96). 1996 Aug 4–8; Portland, Oregan, USA. Menlo Park, CA: AAAI Pr. p 725–30.

Pomeroy SL, Tamayo P, Gaasenbeek M, Sturia LM, Angelo M, McLaughlin ME, Kim JYH, Goumneroval LC, Black PM, Lau C et al. 2002. Prediction of central nervous system embryonal tumour outcome based on gene expression. *Nature*, 415:436–42.

Shipp MA, Ross KN, Tamayo P, Weng AP, Kutok JL, Aguiar RCT, Gaasenbeek M, Angelo M, Reich M, Pinkus GS et al. 2002. Diffuse large B-cell lymphoma outcome prediction by gene-expression profiling and supervised machine learning. *Nature Med*, 8:68–74.

Singh D, Febbo PG, Ross K, Jackson DG, Manola J, Ladd C, Tamayo P, Renshaw AA, D'Amico AV, Richie JP et al. 2002. Gene expression correlates of clinical prostate cancer behavior. *Cancer Cell*, 1:203–9.

Tan AC, Gilbert D. 2003. An empirical comparison of supervised machine learning techniques in bioinformatics. In Proceedings of the First Asia Pacific Bioinformatics Conference. 2003 Feb 4–7; Adelaide, Australia. Sydney: Australian Computer Society. CRPIT 19:219–22.

Ting KM. 1994. Discretization of continuous-valued attributes and instance-based learning. Technical report 491. Available from the University of Sydney, Australia.

van't Veer LJ, Dai H, van de Vijver MJ, He YD, Hart AAM, Mao M, Peterse HL, van der Kooy K, Marton MJ, Witteveen AT et al. 2002. Gene expression profiling predicts clinical outcome of breast cancer. *Nature*, 415:530–6.

Welsh JB, Sapinoso LM, Su AI, Kern SG, Wang-Rodriguez J, Moskaluk CA, Frierson HF Jr, Hampton GM. 2001. Analysis of gene expression identifies candidate markers and pharmacological targets in prostate cancer. *Cancer Res*, 61:5974–8.

Witten IH, Frank E. 2000. Data mining: practical machine learning tools and techniques with java implementations. San Francisco: Morgan Kaufmann.