Combining Heterogenous Classifiers for Stock Selection

George T. Albanis and Roy A. Batchelor
City University Business School, London
September 1999

Abstract

Combining unbiased forecasts of continuous variables necessarily reduces the error variance below that of the median individual forecast. However, this does not necessarily hold for forecasts of discrete variables, or where the costs of errors are not directly related to the error variance. This paper investigates empirically the benefits of combining forecasts of outperforming shares, based on five linear and nonlinear statistical classification techniques, including neural network and recursive partitioning methods. We find that simple "Majority Voting" improves accuracy and profitability only marginally. Much greater gains come from applying the "Unanimity Principle", whereby a share is not held in the high-performing portfolio unless all classifiers agree.

Keywords: Forecasting; Stock Market; Combining Forecasts; Statistical Classification; Discriminant Analysis; Neural Networks; Recursive Partitioning.

Contact Address: Professor Roy Batchelor, City University Business School, Department of Banking and Finance, Frobisher Crescent, Barbican Centre, London EC2Y 8HB.

email: R.A.Batchelor@city.ac.uk

1. Introduction

This paper assesses the value of combining results from statistical classifiers, applied to the problem of identifying high-performing shares on the London Stock Exchange. The classifications are based on financial ratios drawn from the annual accounts of around 700 companies, and predictions are made for the performance of these companies' shares over 1-year holding periods during the years 1993-7. The data are screened using linear discriminant analysis, and four nonlinear techniques – a probabilistic neural network, a vector quantization procedure, an oblique recursive partitioning method, and a rule induction algorithm.

The selection of potentially outperforming shares is the central task of fund management, so our findings may be of direct interest to investment institutions. The stock selection problem also has a number of characteristics that make it interesting from a research perspective. Combining unbiased forecasts of a continuous variable like company earnings necessarily reduces the mean square error below the error of a typical individual forecast (see, for example, Bates and Granger, 1969). There is now a large literature measuring these benefits and discussing weighting schemes for combining forecasts, much of it surveyed in Clemen (1969) and Diebold and Lopez (1995).

However, the target variable in stock selection is not continuous – a share is either in or out of the portfolio - and this requires different combining methods based on voting rules. It is not obvious that such rules will necessarily help. For example, most classification methods will correctly classify "easy" cases, but only a few will succeed with "difficult" cases, which are therefore liable to be filtered out by a majority vote. Moreover, the relevant measure of success in stock selection is not accuracy but profitability, preferably adjusted for risk. For variables such as stock returns, which are significantly non-normal, the returns from trading on point and directional forecasts are known to be only weakly related to their accuracy (Leitch and Tanner, 1990). So the most profitable way to combine any forecasts, and especially binary forecasts, is essentially an empirical matter.

The second section of the paper below sets out the principles that have informed our methodology for classifying share returns. The third section reviews the classification methods we have used. The fourth section introduces the data. The fifth section discusses methods of combining forecasts, and presents our empirical results.

We find that all the classifiers can identify high and low performing stocks ex ante, to a statistically significant degree, with a typical "hit-rate" of around 60%. This translates into an average return on the portfolio of high-performing shares some 6-7% per annum above the 12% return to an equally-weighted stock index. There is little difference in statistical performance and raw profitability across

the different classifiers. Adjusting returns for market risk does reveal some differences, however, with the recursive partitioning method outperforming discriminant analysis and the probabilitistic neural network.

If the portfolio is restricted to shares classed as high by *all* the models, the average return rises to 12-13% above the index. This sharp increase is achieved with lower transactions costs, since a much smaller number of shares is traded. The actual and predicted high-yielding shares are riskier than average. Adjusting for market risk reduces excess returns from the combined classifier to around 9% per year, but this is still better than the best individual classifier. Our conclusion is that combining classifiers is a potent source of excess returns in stock selection tasks. We discuss possibilities for further refining the forecasts in the final section of the paper.

2. Methodology

The predictability of stock prices has been extensively researched over the past few decades. As a result, we feel that certain general principles have now been established, and these have determined our choice of forecasting method. The principles are as follows.

First, the efficient market theory has proved very robust (Fama, 1961; Fama, 1991). The central proposition of the theory is that, because of the activity of profit-seeking speculators, expected returns on shares will reflect their risk. It may be possible to identify in advance which shares will produce returns higher than the general market index, but these shares will be riskier. It may also be possible predict whether one year will be better than another for the stock market as a whole. But the high-earning years will also be years of high risk, or high risk-aversion on the part of investors. In practice the risks attaching to individual shares, and changes in market risk, are themselves hard to predict, so that it is hard for forecasters to beat a random walk model of stock prices, and hence hard for investors to find *any* portfolios which consistently beat the market index.

Second, there is nonetheless evidence of some predictability in stock prices beyond what might be expected on the basis of the efficient market hypothesis. A series of studies has shown that some characteristics of the financial structure of the company can predict returns. Fama and French (1995) and others have suggested that company fundamentals like market capitalisation, and the ratio of the market value of the firm to the book value of its assets, are (inversely) correlated with excess returns. A series of studies has also argued that the time series of share price movements is not truly random. For example, Lo and McKinlay (1988) argue that recent price trends tend to persist into the near future

but reverse in the distant future, and Brock et. al. (1992) find that moving average trading rules used by technical analysts yield excess profits.

Third, these kinds of patterns have proved unstable over time. Small companies and "value" shares seem to perform well in recovery years, and in years of high inflation and low real interest rates. But they do badly in recession years, and in the recent period of low inflation (Fama and French, 1996). One corollary of this is that if rules exist, they are probably complex. There is now an industry looking for more complex "style rotation" relationships, between share returns, accounting ratios, price momentum, and the state of the economy (see, inter alia, Arnott and Copeland, 1985; Sharpe, 1992; Asness, 1997). A second corollary is that if rules can be extracted from data, most weight should be given to recent data. This is a conclusion of most empirical studies of forecasting accuracy in the broader domain of business forecasting, notably Makridakis et. al. (1982). A third corollary is that statistical methods should be preferred to judgmental methods in searching for and applying rules. The judgments of fund managers in this complex environment are notoriously unreliable. The average mutual fund underperforms the index, and there is little evidence that any fund can perform consistently above average (Jensen, 1968; Ippolito, 1989).

Fourth, it is worth combining the forecasts from a number of unbiased forecasting methods (Bates and Granger, 1969). In conventional forecasting tasks, the gains have proved substantial. Combined time series forecasts outperform all of their components in many of the business forecasting experiments in Makridakis et. al. (1982), and consensus forecasts of economic aggregates typically outperform about 70-80% of the constituent forecasts (McNees, 1992). Batchelor and Dua (1995) show that these gains are greatest if heterogeneous forecasters are combined.

This interest in combining has in recent years extended to classification problems, with investigations of the power of "committees of neural networks", and "mixtures of experts" (see Breiman et. al., 1982; Hansen and Salamon, 1990; Xu et. al. 1992; Krogh and Vedelsby, 1995; Heath et. al. 1996). Results from combining classifiers have not been uniformly encouraging. However, these studies usually combine homogeneous models - an example is the use of bootstrap aggregation or "bagging" and "boosting" (Breiman, 1994) to generate sets of neural networks with slightly different architectures. Our exercise is very different and potentially more productive, in that we are combining heterogeneous models, rather than closely related members of the same family of classifiers.

Finally, it is important to control trading costs and price slippage. Many of the profits which appear to be generated by statistical trading rules are eaten up once trading costs are taken into account (see Brock et. al., 1992 and Pesaran and Timmerman, 1995), and rankings of net returns to mutual funds

tend to be inversely correlated with their operating costs. All this argues for investment rules that involve infrequent trading, in a relatively small number of liquid stocks.

3. Classification Methods

The purpose of a classification method is to predict the class to which some new observation belongs, based on data on attributes of known members of possible classes. In our case, we are interested in whether a particular share should be classed as potentially high-return (H), or not (L). Our data are the past returns on a sample of shares, and associated "fundamental" information on the financial state of the companies issuing these shares. We prefer this to the more conventional approach of making point predictions of the expected return to shares, on two grounds. First, having a fuzzy, qualitative, target variable respects the fragility of the relationships between share returns and fundamental factors discussed above. Second, from the viewpoint of an investor, the primary requirement is a simple buy or sell decision.

Classification problems arise naturally in many branches of the natural and social sciences. The traditional approach to solving such problems is the linear discriminant analysis developed by Fisher (1936), and this is the first technique which we apply below. However, advances in theory and computing power over the past two decades mean that a huge array of alternative techniques is now available, many surveyed in Hand (1997). Some of these are statistical, but semi- or non-parametric, and potentially capable of uncovering nonlinear structures in data. In our study these are represented by learning vector quantization, and the probabililistic neural network (a kernel density estimator). Some modern classification methods depend on the induction of logical rules, represented in our study by a recursive partitioning algorithm, and the RIPPER rule induction algorithm. Michie et. al. (1994) conduct a forecasting competition which shows that although some computational algorithms are more efficient than others, all types of classification method have something to offer, depending on the nature of the data.

- *Linear Discriminant Analysis(LDA)*

Suppose y_{it} is the 1-year-ahead return on some share i bought at time t, and $\mathbf{x}_{it} = [\ x_{1it}\ x_{2it}\ \ldots\ x_{nit}\ ]$ the vector of standardised attributes for company i which are known at t. Based on rankings of returns in excess some benchmark index, we assign returns y_{it} to class C_{it} = H or L, depending on whether or not returns exceed some threshold percentile. The problem is to predict the class of y_{it} given current information $\mathbf{x}_{it}$ on company i, and information on the classes C_{jt-k} and related information $\mathbf{x}_{jt-k}$ of all

shares in some earlier "training set" of data. The success of the classification method will be judged by tracking the returns to the portfolio consisting only of shares classed as H.

The idea of linear discriminant analysis is to search for a rule of the form

$$C_{it} = \text{H or L according as } z_{it} = w_0 + \mathbf{w}'\mathbf{x}_{it} > 0 \text{ or } \leq 0 \qquad (1)$$

The parameter vector $\mathbf{w} = [w_0, w_1, \ldots, w_n]$ is chosen so as to maximise $D = (\mathbf{w}'\mathbf{x}^H - \mathbf{w}'\mathbf{x}^L)/ \mathbf{w}'\mathbf{S}\mathbf{w}$, where $\mathbf{x}^H$ and $\mathbf{x}^L$ are the means of the high and low performing groups in the training set, and $\mathbf{S}$ is the estimated (common) within-group covariance matrix. In financial applications, z_{it} is often called the "z-score" of observation $\mathbf{x}_{it}$.

The LDA rule can be interpreted alternatively as assigning a quasi-likelihood f^H_{it} to observation $\mathbf{x}_{it}$ conditional on an elliptical distribution centred on the mean of observations classed as H in the training set, and a corresponding likelihood f^L_{it} from a distribution fitted to the L observations. Observation $\mathbf{x}_{it}$ is then classed as

$$C_{it} = \text{H or L according as } f^H_{it} > f^L_{it} \text{ or } \leq f^L_{it} \qquad (2)$$

That is, a new observation with unknown class is assigned to either H or L, depending on which distribution gives it higher likelihood.

The LDA model provides a simple and fast algorithm that has been applied to a variety of problems in finance, including bankruptcy and bond rating prediction (Altman, 1972, Kaplan and Urwitz, 1979). But the model is restrictive in that it requires the independent variables to follow an elliptical multivariate distribution, such as the Gaussian. Financial ratios of the kind used here are typically highly skewed, flat, and/or dominated by outliers (Deakin, 1976; Frecka and Hopwood, 1983). For this reason, it is worth looking at more flexible alternatives.

- *Learning Vector Quantization (LVQ)*

The Learning Vector Quantization procedure (Kohonen, 1988) assumes that the distribution of each class can be approximated not by a single elliptical distribution, but by a mixture of distributions. Figure 1 illustrates the idea for data with only two attributes, x_1 (say, book-to-market ratio) and x_2 (say size). In all, 6 distributions have been superimposed on the scatter of data points, 3 centred on distinct clusters of

H (□) observations, and 3 centred on clusters of L (●) observations. The ellipses around the centre of each cluster join points with equal likelihood in that cluster, or strictly equal distance from the cluster mean. The space is partitioned into H and L regions by lines connecting points of equal distance from neighbouring means, a "Voronoi tessellation". With only 2 clusters, this model collapses to the LDA, with a single straight-line discriminant function along which z = 0. As the number of clusters is increased, the model becomes more general, and the frontier between H and L need not be linear, as shown by the firm line in the Figure. Indeed all the H cases need not even be in contiguous clusters.

This is equivalent to k-means cluster analysis in a "supervised" form where the classes to which the data clusters belong are known in advance. The Kohonen LVQ algorithm is a computationally convenient way of implementing the supervised k-means model, though not necessarily the most accurate (Balakrishnan et. al., 1994). In the LVQ algorithm, a small number k of prototype "codebook vectors" are identified by sampling in the training set, and these are taken as provisional centres of the distributions. Each element of the training set is compared in turn with the existing codebook vectors, and assigned to the nearest. The mean of that distribution is then moved closer to or further from the new element, depending on whether the new element is or is not of the same class as the nearest codebook vector.

Formally, suppose the new element $\mathbf{x}_{it}$ is the s-th observation in the training set, and the nearest codebook vector is, after considering s-1 training set elements $\mathbf{p}_j\{s-1\}$. Let $\delta_s = 1$ if x_{it} is the same class as $\mathbf{p}_j$ and $\delta_s = -1$ if it is a different class. Then the codebook vector is adjusted as

$$\mathbf{p}_j\{s\} = \delta_s\lambda_s \mathbf{x}_{it} + (1 - \delta_s\lambda_s) \mathbf{p}_j\{s-1\} \qquad (3)$$

That is, the existing mean is weighted together with the new vector, with weights corresponding to some empirically determined "learning rate" λ_s. This process is continued for all of the training set data. If necessary, the whole cycle can be repeated, until a stable set of classes emerges.

Algorithms of this kind have been little applied in finance, though the LVQ performs relatively well in the benchmarking studies of Michie et. al. (1994) which does include data sets on bank lending decisions. In more complex versions, the learning rate may depend inversely on the number of observations already in the class, and in the adaptive k-means model of Dubes and Jain (1976), the Voronoi tessellations are replaced by soft transitions, by explicitly defining a Gaussian kernel at the centre of each cluster. Both the means and variances are then adjusted as the algorithm steps through the training data.

- *Probabilistic Neural Network (PNN)*

The probabilistic neural network represents an extreme generalisation of the k-means model along these lines. In the PNN, the number of clusters k is set equal to n, the training sample size. So whereas linear discriminant analysis positions two elliptical distributions over the means of the two groups, and the k-means model positions k distributions over the data, the probabilistic neural network starts by positioning separate distributions, which may take any reasonable form, over *every* data point. A new observation is then assigned to one class or the other depending on which set of distributions is closer, on the basis of a distance criterion like (2).

Specifically, suppose there are m^H observations in class H. The distance of new observation $\mathbf{x}_{it}$ from the H class is simply the average of distances from all the individual members $\mathbf{x}_{jt-k} \in$ H, as

$$d^H(x_{it}) = \frac{1}{m^H} \sum_{j \in H} f(x_{it}, x_{jt-k}) \tag{4}$$

The most popular choice for the distance measure is the Gaussian function

$$f(x_{it}, x_{jt-k}) = \frac{1}{\sqrt{2\pi}.\sigma} \exp\left(- \frac{\left\| x_{it} - x_{jt-k} \right\|^2}{2\sigma^2} \right) \tag{5}$$

This procedure gives a kernel density estimator for the class of H shares, using a multivariate kernel with a mixture of identical Gaussian distributions positioned at each of the training sample points (Hand, 1997, p 85). This PNN has been applied to problems of financial distress and bond ratings prediction (Tyree and Long, 1997; Albanis and Batchelor, 1999a), with encouraging results.

The benefit of the PNN is that does not require multivariate normality or equality of the group covariance matrices, and in that sense is much less restrictive than LDA. And it does not restrict the number of segments in the tessellation, so it can model more complex discriminant surfaces than the LVQ. The problem with the PNN, as with any kernel estimator, lies in the selection of an appropriate smoothing parameter σ. A low value of σ means that the distance measures are heavily influenced by the nearest training set data. This has the benefit that complex features of the distributions will be captured, but it means classification may be unduly influenced by idiosyncrasies in the sample, or

measurement error in the input vector. Choosing a larger value for σ reduces these problems, but makes the distributions closer to those assumed by LVQ and LDA.

The "neural network" label is due to Specht (1990), who noted that the computational algorithm could be characterised by the four-layer architecture of Figure 2. The input layer consists of the n elements of the vector $\mathbf{x}_{it}$. The pattern layer consists of as many nodes as there are observations in the training set (m, say). At each node of the pattern layer, the potential function (5) is evaluated. In the summation layer, there are two nodes, corresponding to the two possible classes. The distance measures for the H class are summed in one of these nodes, as in Equation (4) above, and the distance measures for the L class in the other. The output node assigns the input vector to the H or L class, depending on which of these distance measures is smaller, the prior probabilities of observing H and L, and relative costs of classification errors.

- *Recursive Partitioning (Oblique Classifier OC1)*

Recursive partitioning methods, or decision trees, approach the problem of classification in a way quite distinct from the semi-parametric methods outlined above. The idea is to find an ordered sequence of binary conditions for elements of the x_{it} that will lead to a correct decision for new observations. Figure 3 shows a plausible decision tree for classifying the data of Figure 1 into H and L classes, and Figure 4 shows the corresponding partitioning of (x_1, x_2) space. In this example two of the conditions are "axis-parallel" – that is, they compare only one of the variables with a threshold value. The final condition is "oblique", and compares a linear function of the variables with a threshold value.

In general, the method consists of finding a sequence of rules of the form

$$\textit{if } \{z_{it} = w_0 + \mathbf{w}'\mathbf{x}_{it} > 0\} \textit{ then } \{\text{next}^H \textit{ or } C_{it} = H\}, \textit{ else } \{\text{next}^L \textit{ or } C_{it} = L\} \qquad (6)$$

Where next^H and next^L represent the criteria at the next nodes in the tree. The computational problem is to find reasonable values for the parameters w_0 and $\mathbf{w}$ defining the separating hyperplane, and indeed to find a point at which to stop building the tree. With m-1 criteria, it would be possible to correctly classify all m observations in the training data, but such a tree would be unlikely to generalise. The usual approach to these problems is to search for rules that maximise the change in "impurity" at each node of the tree, and to stop building the tree when this change becomes small. One

plausible measure of impurity at a node is the probability that observations reaching that node belong to the minority class at the node, and hence are potentially classified incorrectly.

For the data in Figure1, there are 30 H cases and 30 L cases, so a sample-based estimate of the probability of correctly classifying, say, an H observation arriving at Node 1 in Figure 3 is 1/2. After applying the rule *if* $\{z = 0.3 - x_1 > 0\}$ *then* C_{it} = H, the impurity at Node 2 is zero, since all 15 cases arriving at this mode are correctly classified as H. However at Node 3 there are 30 ("correct") L cases, but 15 incorrectly classified H cases, so the impurity measure is 15/ (30 + 15) = 1/3. Impurity is reduced in moving from Node 1 to Nodes 2 and 3, and hence the criterion at Node 1 is worth applying.

This is one of many impurity measures, and suffers from a number of drawbacks, discussed in Hand (1997, p 68). In our work below we use instead the "twoing rule" proposed in the standard text on recursive partitioning algorithms by Breiman et. al. (1984). Suppose we represent the numbers of outcomes from a node as a two-way contingency table:

	$z > 0$	$z < 0$
Actual = H	m_{11}	m_{12}
Actual = L	m_{21}	m_{22}
Sum =	$m_{.1}$	$m_{.2}$

Here $m_{.1}$ is the total number of cases on the left of the split, m_{11} is the number of cases on the left of the split classed as H, and the total number of observations considered at the node is $m = m_{.1} + m_{.2}$. The twoing criterion is *min* (1/M), where

$$M = \frac{m_{.1}}{m} . \frac{m_{.2}}{m} . \sum_{j=1,2} \left[\frac{m_{j1}}{m_{.1}} - \frac{m_{j2}}{m_{.2}} \right]^2 \qquad (7)$$

We use the OC1 algorithm of Murthy et. al. (1994) to search first for the best axis-parallel split at a node, and then for oblique splits. As with any numerical procedure, there is a danger that these searches will yield local rather than global minima for the impurity measure. After finding the best split at each node, the algorithm makes random perturbations to the coefficient vector and compares the resulting impurity measure with the selected rule.

- *RIPPER Rule Induction (RRI)*

Rule induction methods are closely related to recursive partitioning. Like the decision tree, rule induction involves searching for a sequence of logical conditions that have a high probability of separating the H and L classes. For the data of Figures 1 and 4, for example, these conditions might be:

H if ($x_1 < 0.3$ and $x_2 > 0.3$ and $x_2 < 0.6$)
H *if* ($x_1 < 0.4$ and $x_2 < 0.3$)
H *if* ($x_1 < 0.7$ and $x_2 > 0.7$)
Else L

This could be expressed as a decision tree or recursive partitioning with axis-parallel splits, and in that sense the method is less general than the OC1 classifier described above. One benefit claimed for using explicit rules is that they may have a commonsense interpretation, which is not true of most of the above methods. Another benefit is that rules can be applied to non-numeric or fuzzy variables, though this is not relevant to our data. In the most common applications of rule induction, in text processing and massive database searches, computational speed is at a premium, and rule induction algorithms are relatively efficient. Finally, there is less danger of overfitting with axis-parallel rules. In a study of US corporate bankruptcy, Frydman et. al. (1985) found a recursive partitioning method with only axis-parallel splits produced significantly better out-of-sample predictions than linear discriminant analysis.

The most popular rule induction techniques start by developing an excessively complex rule, and then simplify it on the basis of the predictive performance of the rule. The method we use is the RIPPER algorithm due to Cohen (1995). This involves splitting the training data into two sets - a *growing set*, on which the overly complex rules are developed, and a *pruning set*, which is used to decide which elements of the complex rule can safely be discarded. The growing set is usually taken to be a random sample of the training data. The first step is to find a condition of the form:

$$\text{If}(\ x_1 \otimes v_1 \text{ and } x_2 \otimes v_2 \text{ and .. and } x_n \otimes v_n) \text{ then H} \qquad (8)$$

which holds without exception in the growing set, where $\otimes$ is one of the operators $<, \leq, >, \geq, =$, and the v_i are possible values for the input variables. The complete rule is "grown" by adding one conjunctive element after another, until no element can be added to the rule without misclassifying at least one observation. Of course, not all the x_i need be involved in the condition. This condition is then applied to the pruning set, and the proportion of successful classifications noted. Each element of the rule is then dropped in turn, until the success rate in the pruning set is maximised. The pruned version of the rule is retained. Importantly, all observations covered by the rule are then dropped from the training set.

A new rule is then sought to cover the H cases which remain in the growing set, and the process is repeated. This process of postulating general rules and pruning them is continued until no progress can be made in classifying the pruning set.

4. Data and Trading Rules

Our target data are total returns on all shares traded on the London Stock Exchange in the years 1993-97 for which there is a complete set of annual accounts available on the *EXTEL* service. In all, 651 shares met this criterion in 1993, rising to 752 in 1997. Returns are measured on an annual basis, using end-month price data, and assuming reinvestment of any dividends received during the year. Excess returns are calculated for each share by subtracting the corresponding total return on an equally weighted index of all sampled shares from the individual share return. The returns data all come from *Datastream*.

For each year, the excess returns from one-year investments starting at the end of all months in that year are ranked. A share is classified as "High-Performing" (H) in that year if its return is in the top 25% of this ranking. Otherwise it is classified as "Low-Performing" (L). The 25% cut-off is selected so as to give a clear difference between the mean returns on the H and L groups, while keeping the group sizes reasonably large.

To predict whether a particular share will be H or L, we use information in the companies published annual accounts for the previous year. From these we extracted information on 15 key balance sheet items - sales revenue, earnings, total profits, tax paid, total assets, total liabilities, current assets, current liabilities, current debtors, total capital employed, shareholders equity, dividends paid, market capitalisation, book value of assets, and total debt. These were then used to form 38 conventional financial indicators, such as profits after tax relative to sales, price to earnings ratio, book to market ratio, and earnings per share.

Our aim is to find rules that will classify a particular share as H or L based on these indicators. Consistent with our principle of giving most weight to recent data, we restrict our information set to share returns in the previous two calendar years. Thus data for 1991 and 1992 are used to classify shares in the 12 months of 1993; data for 1992 and 1993 are used to classify shares in 1994; and so on. If a share is classified as H, we assume that it is bought at the end of the reporting month, and held for one year. Equal amounts of each share are bought, regardless of the strength of the trading signal. The value of each classification method is judged by the cumulative profits generated by the resulting rolling portfolio of H shares.

The benefit of this rule is that it minimises transactions costs, and is not affected by any anomalies in price behaviour around the reporting date. Each share is traded at most once per year, and trades can be done in a basket on one day at the end of each month. On average, the H portfolio will turn over 1/12 of its constituents each month. Given that there are 160-190 H shares each year, only 13-16 shares will be bought and sold each month in the ideal trading strategy. A possible disadvantage of our trading rule is that there is a delay of up to 4 weeks in reacting to the accounting news. Another is that we do not invest relatively heavily in shares which some rules may indicate have a very high probability of success. And we do not "manage" the portfolio in any way, for example by selling underperforming shares during the year.

In practical applications of all classifiers, three generic problems need to be addressed. One concerns the normalisation of the inputs. The second concerns the relative costs of misclassifying H and L shares. The third concerns the potential redundancy of elements of the input vector. Predictions for 1993, made using data from 1991-2, were used in experiments to determine how best to handle these problems. The year 1993 cannot therefore be regarded as truly "out-of-sample", and results for 1993 and 1994-7 are reported separately below.

Classification algorithms work best if the data is spread evenly across the input space. As noted earlier, financial ratios are typically highly nonnormal, and we have many series with extreme outliers. For the current exercise, we have winsorized the data for each year, by ranking all the observations on each series, and setting values in the lower and upper 5% tails equal to the 5^{th} and 95^{th} percentile values. The results are then squeezed into the range [0, 1] as $x_{kit} = (r_{kit} - min_{kT})/(max_{kT} - min_{kT})$, where r_{kit} is the winsorized ratio, and max_{kT} and min_{kT} are respectively the highest and lowest values for variable k in the year T.

Classification rules like (1) and (2) assume that the costs of misclassification are the same for both groups, and that we have no priors about the probability of observing each class. However, in general (2) should be

$$\text{Cit} = \text{H or L according as } \frac{\pi^H}{\pi^L}.\frac{c^H}{c^L}.f_{it}^H > f_{it}^L \text{ or } \leq f_{it}^L \qquad (9)$$

where π^H and π^L are prior probabilities of observation $\mathbf{x}^{it}$ lying in the H and L classes, and c^H and c^L are the costs of incorrectly classifying shares which belong to the H and L classes. It is hard to guess at the costs of misclassification, since this depends on the distribution of returns over the two classes. However, we can assign reasonable priors. If the stock market is efficient, returns available to investments made at the end of each month should be independent of information received before the end of the month, and in

particular the accounting ratios $\mathbf{x}_{it}$. Hence the probability that any randomly selected share is H would be 0.25, and the probability that any share is L would be 0.75. This efficient market prior has been used directly in implementing the LDA, LVQ and PNN and, by requiring proportionately lower impurity on H branches of the decision tree, in the OC1 algorithm.

Classification rules work best with a small set of near-orthogonal inputs. With 38 inputs, many measuring similar aspects of company performance, there is an obvious danger that a model will be overparameterised. We have reduced the dimension of the input data set for all classifiers by variable deletion. In the case of LDA, we apply a stepwise variable selection algorithm, with variables entered one at a time on the basis of their F-statistics, and with all variables already included in the model checked at each stage in case they cease to be significant. The predictive power of the model is judged at each step using full cross-validation in each 2-year training set. For the other classifiers, there is no analytical criterion for variable selection, and we have followed a "general-to-specific" strategy. Again using only data from the training set, each model is implemented using all inputs, and the misclassification rate noted. The variables are then dropped one after another, and the models re-estimated. Full "leave-one-out" cross-validation is used to evaluate the resulting PNNs, but the computational time involved in of applying this to the other methods is prohibitive, and we have instead made 10 "leave 10% out" passes through each training set. The 1991-2 training set gave models to predict 1993 with 12, 19, 17, 21 and 35 variables respectively for the LDA, LVQ, PNN, OC1 and RRI, with similar results for later years. In Albanis and Batchelor (1999b) we discuss alternative data reduction techniques based on nonlinear principal components.

5. Individual versus Combined Forecasts

Each classifier j can be thought of as producing a prediction value I_j of 1 or 0 for an out-of-sample observation, according as the company is predicted to have a High or Low share price performance. A composite linear prediction rule which encompasses much recent research assigns an observation to H if

$$V = \sum_{j=1}^{n} w_j I_j \geq V^*$$

where n is the number of classifiers (in our case 5), and the w_j are weights on each prediction, which without loss of generality can be made to sum to unity. Mani (1991) shows that some voting rule of this kind will necessarily improve accuracy as the number of independent unbiased classifiers increases. The effectiveness of combining procedures will depend on exactly how the weighting is done, and how the voting is done – that is, on choices of w_j and V*.

In business forecasting, experience suggests that most benefits from combining are captured by equally-weighted schemes, with $w_j = 1/n$ (see Diebold and Lopez, 1995). In artificial intelligence applications, the empirical evidence is unclear. For example, Alpaydin (1998) finds that for nearest neighbor classifiers like the PNN, significant improvements in accuracy can be achieved by weighting according to the models' performance on subsets of training data. But Ali and Pazzani (1995) test weights based on likelihoods for partitioning rules like our OC1 and RRI methods on a large number of disparate data sets, and prefer voting based on equal weights.

In combining classifiers, the most commonly used values for V* are n/2, the Weighted Majority Rule, and n, which requires a Unanimous Vote. Again, it is unclear which is likely to work best in practice. Some empirical studies, such as Battiti and Colla (1994), find the majority vote outperforms the unanimous voting rule. Others, like Heath et. al. (1996) favour unanimity.

Because the in-sample performance of the five classifiers in our exercise is broadly similar, we have not investigated weighting schemes here. However, we do compare the effects of a rule that classifies a share as H if a majority of classifiers favour it (the MVH rule), and a rule that classifies a share as H only if all five classifiers agree unanimously (the UVH rule). The five models and these two combining rules are judged on the percentage of shares correctly classified, and more importantly on the profits generated from trading on the rules.

Table 1 summarises classification rates for all methods, for the target year 1993 – on which the input dimensionality reduction experiments were conducted – and for the genuine out-of-sample target years 1994-7. The LDA with 12 inputs produces excellent results for 1993, with around 70% of shares correctly classified, and about 60% of high-performing shares correctly classified. Given that the top 25% of shares are assigned to the high performing class, this is well in excess of what might be expected by chance. The PNN produces comparable results. The other methods, although using more inputs, classify about 65% of shares correctly.

With the same models applied to 1994-7, overall performance degrades a little, but remains far better than chance for all classifiers. Figure 5 shows that the general level of performance is stable over the years 1994-7, though there are small changes in the rankings of the individual methods from year to year. The LDA performs relatively poorly (57.8% correct), suggesting that our stepwise variable selection criterion was perhaps too parsimonious. On the other hand the RRI performs even worse (56%). Since the RRI retained most of the input variables, and for each year generates about 18-20 rules, the more complex of which have little predictive power, this suggests that the RRI model is not parsimonious enough. The other nonlinear methods generally perform better than the LDA on the 1994-7 data, and

only a little worse than in the benchmarking year 1993, with the PNN, LVQ and OC1 all classifying around 60% of shares correctly.

In spite of our strong prior against classifying a share as high-performing, a substantial number of L shares are incorrectly classified as H. For the LDA, almost 70% of shares classed as H are actually L, and for the best classifier (OC1) the proportion is 62%. This is of practical importance, since our trading rule involves buying all shares classed as H. Whether the contamination matters depends on whether the L shares wrongly classed as H are marginal cases, or whether they are typical low-performing shares.

The final two rows of Table 1 show classification results for the Majority Voting and Unanimous Voting composite classifiers. The Majority Voting rule produces results comparable to the best of the individual classifiers, with about 70% correct in 1993 and about 60% correct in 1994-7. However, the Unanimous Voting rule is markedly superior to all of the other classifiers. In 1993 it has an accuracy rate of nearly 75%, and this outperformance persists into the 1994-7 period, with an overall accuracy of over 70%. The Unanimous Voting rule also produces much smaller H portfolios than the individual classifiers and the MVH rule. The individual methods typically give portfolios of around 300 predicted H shares, but the UVH rule gives portfolios of a core of about 100 shares common to all the methods. Since each share is held for a year, this involves buying and selling only 8-12 shares on one day towards the end of each month.

Table 2 measures the financial returns to an equally weighted index of all shares in our data, to portfolios of the actual H and L shares in all the twelve-month holding periods starting in each year, and to the portfolios predicted to be H and L by our classifiers.

Looking at results for the individual classifiers, two features stand out. First, all produce excess returns. The equally weighted index grew by about 12% per year in the period 1994-7. The H portfolios of all classifiers comfortably outpace this, with the worst (RRI) yielding about 16%, and the best (PNN) 19%. Second, there is some correlation between accuracy and returns. The RRI, which classifies worst produces the lowest profits. The other methods were very similar in terms of accuracy and offer similar returns, in the 18-19% range.

The Majority Voting Rule, which made little difference to accuracy, also makes little difference to profits, outperforming the individual classifiers only marginally. The most striking feature of the table is the performance of the Unanimous Voting rule, which produces a return to the H portfolio of over 25% per annum in the 1994-7 period. Its excellent performance relative to the alternatives is illustrated in Figure 6.

Findings like this are vulnerable to the accusation that high returns are made only at the expense of high risk. Some relevant characteristics affecting risk are summarised in Table 3, which shows averages for the target year 1995 of selected attributes of the actual and predicted H portfolios from the PNN. Portfolios from other classifiers and years are similar. The actual H portfolios consist of companies that have a smaller-than-average market capitalisation, and hence may be riskier and less liquid. The actual and predicted H companies are also typically highly geared (high debt: equity ratio) and show a low current after-tax profit. They tend pay out low dividends, and tend not to be "value" companies, but rather "growth" companies, with high market-to-book ratios.

A more formal adjustment for risk using the Capital Asset Pricing Model is made in Table 4, which shows regression-based estimates of the betas (gearing to the equally-weighted index) and alphas (excess returns not due to this gearing) for actual and predicted H and L portfolios. The actual H portfolio does have a high beta, around 1.4. The predicted H portfolios from the LDA, PNN and UVH have similarly high betas, but the other classifiers generate H portfolios with betas very close to 1. This means that in terms of beta-adjusted excess return (i.e. alpha), the LVQ and OC1 classifiers produce results almost as good as the UVH combining rule. For example, the OC1 H portfolio has a return of 18.9%, but a beta of only 0.9, giving an alpha of 8.1%. The UVH H portfolio has a much higher raw return of 24.8%, but also a much higher beta of 1.39, to give an only marginally higher risk-adjusted return of 9.3%.

6. Concluding Remarks

Our experiments suggest that statistical classification methods can identify ex ante portfolios of shares that will consistently outperform an equally-weighted benchmark index. Linear discriminant analysis is only marginally outperformed by the more complex nonlinear models in terms of accuracy and profitability. Combining these classifiers using Majority Voting makes little difference. But combining using a Unanimous Voting rule markedly improves overall accuracy, overall profitability, and this improvement is achieved with lower trading costs.

Controlling for market risk using the CAPM does not account for all of the excess returns, but it does have an impact on the ranking of the alternative classification methods. The LDA and PNN look less attractive, while the recursive partitioning rule OC1 looks more attractive, in terms of risk-adjusted return. The UVH combining rule still dominates all the individual classifiers and the MVH rule.

These conclusions are of course conditional on the information sets we have used, the way we have implemented the various models, and the trading rule we have assumed. There are some obvious extensions which could be made to the study. For example, in addition to the set of company-specific

attributes considered here, we could look at the more general economic indicators, and patterns in the time series of share prices. We could explore whether it might be better to define the H class as, say, the top 5% or 10% rather than the top 25% of shares. And we could almost certainly improve the return/ risk tradeoff by "managing" the predicted H portfolios, by cutting losses on poorly performing shares before the end of each 12-month holding period. Some of these ideas are explored further in Albanis and Batchelor (1999c). However, our object has not been to seek the best possible trading rule, but to assess the benefits of combining classifiers, and none of these refinements seems likely to undermine the superiority of the Unanimous Voting rule.

References

Albanis G.T., and R. A. Batchelor, 1999a, Using probabilistic neural networks and rule induction techniques to predict long-term bond ratings, *Proceedings of SCI /ISAS Conference Orlando*, 3, 175-80.

Albanis G.T., and R. A. Batchelor, 1999b, Bond ratings prediction: the value of nonlinear dimensionality reduction, *Journal of Computational Intelligence in Finance,* forthcoming.

Albanis, G. T. and Roy A. Batchelor, 1999c, *Economic versus accounting factors in predicting excess stock returns*, Discussion Paper, City University Business School, London.

Altman, E. I., Financial ratios, discriminant analysis and the prediction of corporate bankruptcy, *Journal of Finance,* 23, 589 - 609, 1968.

Ali K.M. and Pazzani M.J., 1995, *Error reduction through learning multiple descriptions*, Dept. of Information and Computer Science Technical Report *95-39*, University of California, Irvine.

Alpaydin E., 1998, Techniques for Combining Multiple Learners, *Proceedings of Engineering of Intelligent Systems Conference*, Vol. 2, pp. 6-12, ICSC Press.

Arnott, R., and W. Copeland, 1985, The business cycle and security selection, *Financial Analysts Journal*, March-April, 26-33.

Asness, C., 1997, The interaction of value and momentum strategies, *Financial Analysts Journal*, March-April, 29-36.

Balakrishnan, P. V., Cooper, M. C., Jacob, V. S., and Lewis, P. A., 1994, A study of the classification capabilities of neural networks using unsupervised learning: a comparison with k-means clustering, *Psychometrika*, 59, 509-525

Batchelor R. A., and P. Dua , 1995, Forecaster diversity and the benefits of combining forecasts, *Management Science,* 41, 68-75.

Bates J. M. Granger C. W. J., 1969, The combination of forecasts, *Operations Research Quarterly,* 20, 451-468.

Battiti R. and Colla A.M., 1994, Democracy in neural nets: voting schemes for classification, *Neural Networks 7* : 691-707.

Breiman L., 1994, *Bagging Predictors*, Technical Report 421, Department of Statistics, University of California, Berkeley.

Breiman L., Friedman J.H., Olsen E.A. and Stone C.J., 1984, *Classification and Regression Trees,* , Belmont, CA: Wadsworth International Group.

Brock W. A. LeBaron B. Lakonishok J., 1992, Simple technical trading rules and the stochastic properties of stock returns, *Journal of Finance,* 47, 1731-1764.

Clemen R. T., 1989, Combining forecasts: a review and annotated bibliography, *International Journal of Forecasting,* 5, 559-581.

Cohen W. W., 1995, Fast effective rule induction, *Machine Learning: Proceedings of the Twelfth International Conference*, Morgan Kaufman.

Deakin, E. B., 1976, Distributions of financial accounting ratios: some empirical evidence, *The Accounting Review*, January, 90 – 96.

Diebold F. X. and J. A. Lopez , 1995, Forecast evaluation and combination, in Maddala G. S. and C. R. Rao (Eds), *Handbook of Statistics Volume 14: Statistical Methods in Finance*, Amsterdam: North Holland.

Dubes, R. and A. K. Jain, 1976, Cluster analysis: the user's dilemma, *Pattern Recognition*, 8, 247-260.

Fama, E. F., 1991, Efficient capital markets II, *Journal of Finance*, 46, 1575-1617.

Fama, E. F., and K. R. French, 1995, Size and book-to-market factors in earnings and returns, *Journal of Finance*, 50, 131-154.

Fama, E. F., and K. R. French, 1996, Multifactor explanations of asset pricing anomalies, *Journal of Finance*, 51, 55-84.

Fisher R.A., 1936, The use of measurements in taxonomical problems, *Annals of Eugenics* 8, pp. 179-184.

Frecka, T. J. and Hopwood, W. S., 1983, The effects of outliers on the cross-sectional distributional properties of financial ratios", *The Accounting Review*, 58, 1.

Frydman, H., E. I. Altman and D-L. Kao, 1985, Introducing recursive classification for financial classification: the case of financial distress, *Journal of Finance*, 40, 269-291.

Hand, D. J., 1997, *Construction and Assessment of Classification Rules*, Chichester; Wiley.

Heath D. , Kasif, S., and Salzberg S., 1996, Committees of decision trees, In Gorayska B. and Mey J. (eds), *Cognitive Technology: In Search of a Human Interface*, Elsevier Science, Amsterdam, The Netherlands. 305-317, 1996.

Hansen L.K. and Salamon P., 1990, Neural Network Ensembles", *IEEE Transactions on Pattern Analysis and Machine Intelligence* 12:993-1001.

Ippolito, R. A., 1989, Efficiency with costly information: a study of mutual fund performance 1965-1984, *Quarterly Journal of Economics*..

Jensen, M. C., 1968, Problems in the selection of security portfolios: the performance of mutual funds in the period 1945-64, *Journal of Finance*, 23,

Kaplan, S. and G. Urwitz, 1979, Statistical models of bond ratings: a methodological inquiry", *Journal of Business*, 52, 2, 231-261

Kohonen, T., 1988, Learning vector quantization, *Neural Networks*, 1, (Supplement)

Krogh A. and Vedelsby, 1995, Neural network ensembles, cross validation, and active learning, in G. Tesauro, D.S. Touretzky, and T.K. Leen (Eds), *Advances in Neural Information Processing Systems 7*, MIT Press, pp. 231-238.

Leitch G., and J. E. Tanner , 1991, Economic forecast evaluation: profit versus conventional error measures, *American Economic Review*, 81, 580-590.

Lo A. , and A. C. McKinlay , 1988, Stock prices do not follow random walks: evidence from a simple specification test, *Review of Financial Studies*, 1, 41-66.

Michie, D., D. J. Spiegelhafter and C. C. Taylor (Eds), 1994, *Machine Learning and Statistical Classification*, Ellis Horwood,

McNees S. K., 1992, The uses and abuses of 'consensus' forecasts, *Journal of Forecasting,* 11, 703-710.

Makridakis S. Andersen A. Carbone, R., Fildes, R., Hibon, M., Lewandwski, R., Newton, J., Parzen, E. Winkler, R., 1982, The accuracy of time series (extrapolative) methods: results of a forecasting competition, *Journal of Forecasting,* 1, 111-153.

Mani, G.,1991, Lowering variance of decisions by artificial neural network ensembles, *Neural Computation*, Vol. 3, pp. 484-486.

Murthy, S. K., Kasif, S. and S. Salzberg, 1994, A system for induction of oblique decision trees, *Journal of Artificial Intelligence Research,* 2, 1-32.

Pesaran M. H., and Timmerman A. , 1994, Forecasting stock returns - an examination of stock market trading in the presence of transactions costs, *Journal of Forecasting,* 13, 335-367.

Sharpe, W. F., 1992, Asset allocation: management style and performance measurement, *Journal of Portfolio Management*, Winter, 7-19.

Specht D., 1990, Probabilistic neural networks, *Neural Networks 3*, pp. 109-118.

Tyree, E. and Long, J.A., 1996, Assessing financial distress with Probabilistic Neural Networks", in A.N. Refenes (ed.) - *Proceedings of the Third International Conference on Neural Networks in the Capital Markets*, London Business School.

Xu L., Krzyzak, Suen C.Y. "Methods of Combining Multiple Classifiers and their Applications to Handwriting Recognition", *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 22, pp. 418-435, 1992.

Table 1. Actual v. Predicted Classification Rates, 1993 and 1994-7

Method	Predicted	Actual 1993			Actual 1994-7		
		H	L	% right	H	L	% right
LDA	H	98	128	*43.4*	409	896	*31.3*
	L	65	360	*84.7*	303	1236	*80.3*
	% correct	*60.1*	*73.8*	*70.4*	*57.4*	*58.0*	*57.8*
LVQ	H	90	154	*36.9*	379	813	*31.8*
	L	73	334	*82.1*	333	1319	*79.8*
	% correct	*55.2*	*68.4*	*65.1*	*53.2*	*61.9*	*59.7*
PNN	H	96	123	*43.8*	390	845	*31.6*
	L	67	365	*84.5*	322	1287	*80.0*
	% correct	*58.9*	*74.8*	*70.8*	*54.8*	*60.4*	*59.0*
OC1	H	88	163	*35.1*	393	804	*32.8*
	L	75	325	*81.3*	319	1328	*80.6*
	% correct	*54.0*	*66.6*	*63.4*	*55.2*	*62.3*	*60.5*
RRI	H	91	149	*37.9*	402	939	*30.0*
	L	72	339	*82.5*	310	1193	*79.4*
	% correct	*55.8*	*69.5*	*66.1*	*56.5*	*56.0*	*56.1*
MVH	H	96	123	*43.8*	399	817	*32.8*
	L	67	365	*84.5*	313	1315	*80.8*
	% correct	*58.9*	*74.8*	*70.8*	*56.0*	*61.7*	*60.3*
UVH	H	32	33	*49.2*	162	265	*37.9*
	L	131	455	*77.6*	550	1867	*77.2*
	% correct	*19.6*	*93.2*	*74.8*	*22.8*	*87.6*	*71.3*

Table 2. Returns from Actual and Predicted Portfolios, 1993-7

Method	Portfolio	1993	1994	1995	1996	1997	1994-7
Actual	*Index*	*30.5*	*5.8*	*25.2*	*9.6*	*9.1*	*12.2*
Actual	*High*	*90.0*	*45.5*	*79.8*	*54.5*	*58.4*	*59.1*
LDA		50.9	8.1	37.8	13.2	16.3	*18.3*
LVQ		42.5	6.6	36.0	16.4	17.1	*18.6*
PNN		51.8	9.1	38.7	13.2	17.0	*19.0*
OC1		35.9	10.5	35.4	15.3	15.8	*18.9*
RRI		43.3	8.2	29.9	14.2	12.0	*15.8*
MVH		50.5	10.2	38.4	15.3	17.0	*19.7*
UVH		58.2	12.3	43.7	21.0	24.3	*24.8*
Actual	*Low*	*9.2*	*-7.6*	*7.1*	*-5.3*	*-7.2*	*-3.4*
LDA		18.1	3.6	14.9	6.3	3.9	*7.1*
LVQ		21.6	4.9	18.1	4.6	3.8	*7.7*
PNN		18.1	3.1	15.3	6.5	3.9	*7.1*
OC1		25.4	1.8	18.4	5.6	4.5	*7.4*
RRI		21.3	3.2	21.0	5.4	7.1	*9.0*
MVH		18.8	2.1	15.6	5.0	4.1	*6.6*
UVH		13.3	2.6	12.6	1.3	1.6	*4.4*

Table 3. Attributes of Actual and Predicted High Portfolios from PNN, 1995

Portfolio	Return %	Size £m	Debt/ Equity	Profit/ Sales	Price/ Earnings	Dividend Yield	Market/ Book
Act High	79.8	266.5	69.8	-10.9	14.1	3.1	167.7
Pred High	38.9	205.9	78.4	-14.4	9.8	3.0	197.7
Pred Low	15.1	731.0	42.0	10.0	20.1	4.1	127.5
Act Low	7.1	586.4	53.5	3.0	16.2	3.9	154.1

Table 4. Betas of Predicted High Portfolios

Method		Return	Alpha	Beta
Actual	*All*	*12.2*	*0*	*1*
Actual	*High*	*59.1*	*43.2*	*1.39*
LDA		18.3	2.9	1.38
LVQ		18.6	5.7	1.11
PNN		19.0	3.5	1.38
OC1		18.9	8.1	0.90
RRI		15.8	3.6	1.10
MVH		19.7	5.1	1.30
UVH		24.8	9.3	1.39

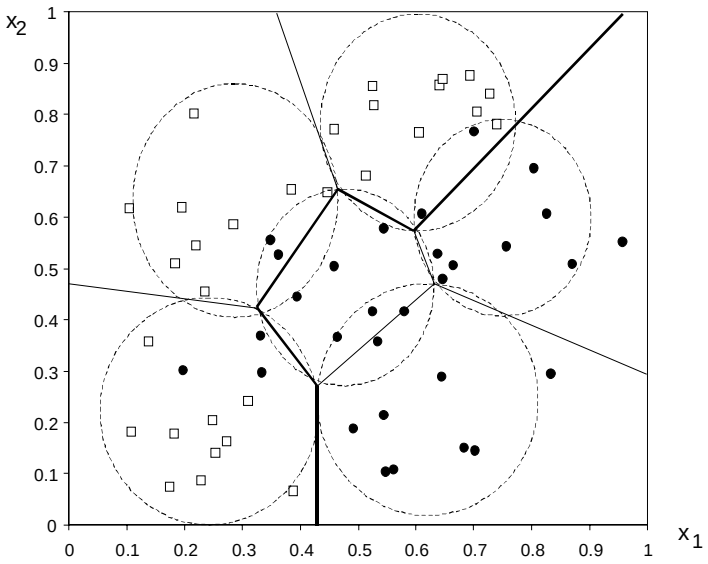

Figure 1. Classification by Learning Vector Quantization

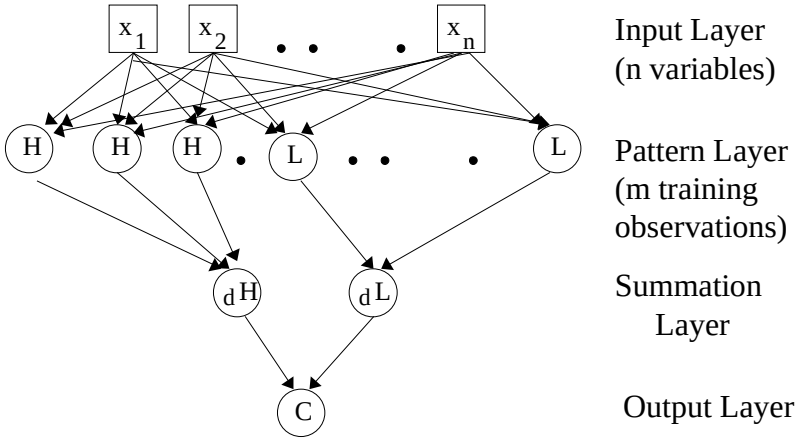

Figure 2. The Probabilistic Neural Network

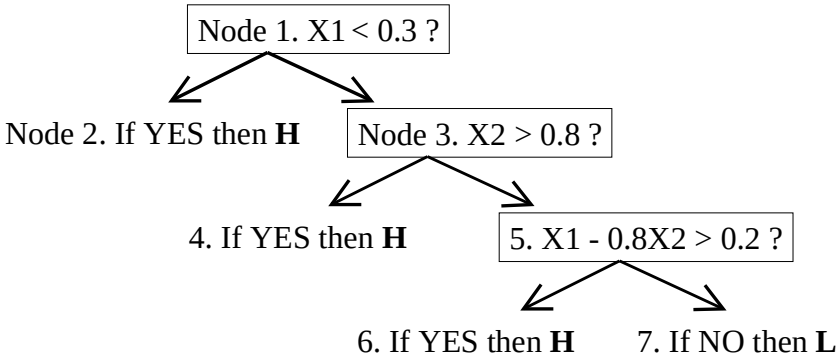

Figure 3. Decision Tree for Share Classification

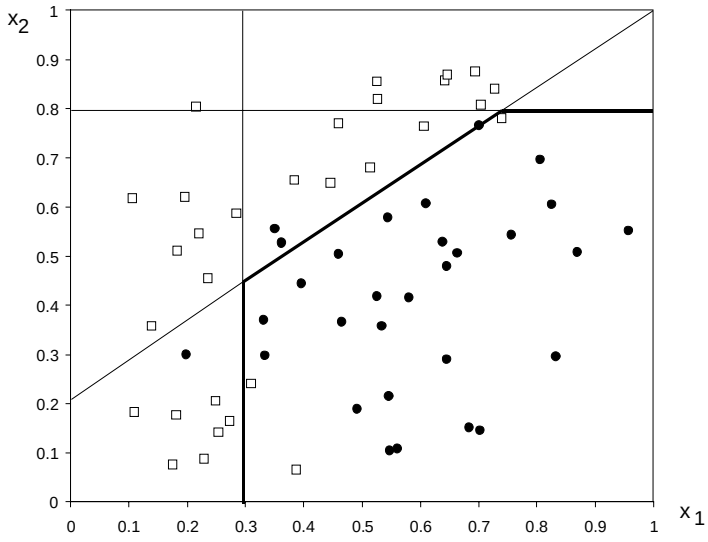

Figure 4. Axis-Parallel and Oblique Recursive Partitioning

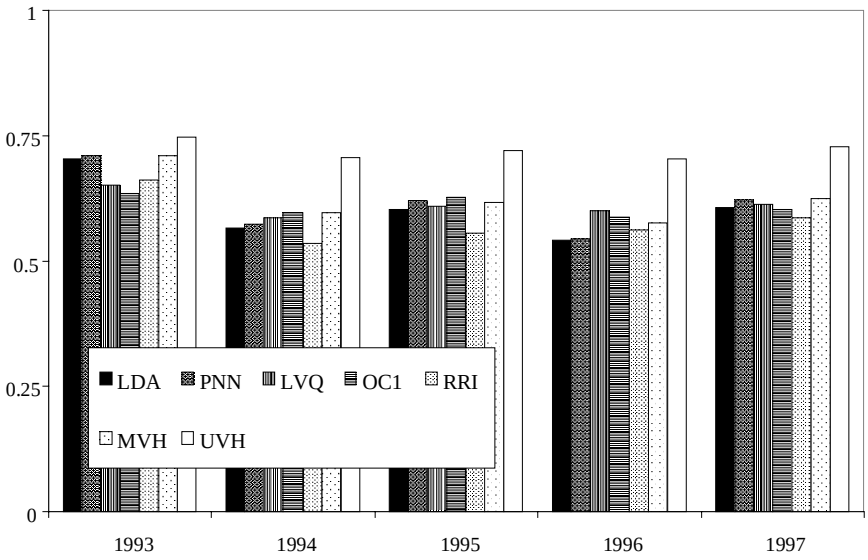

Figure 5. Overall Classification Rates, 1993-7

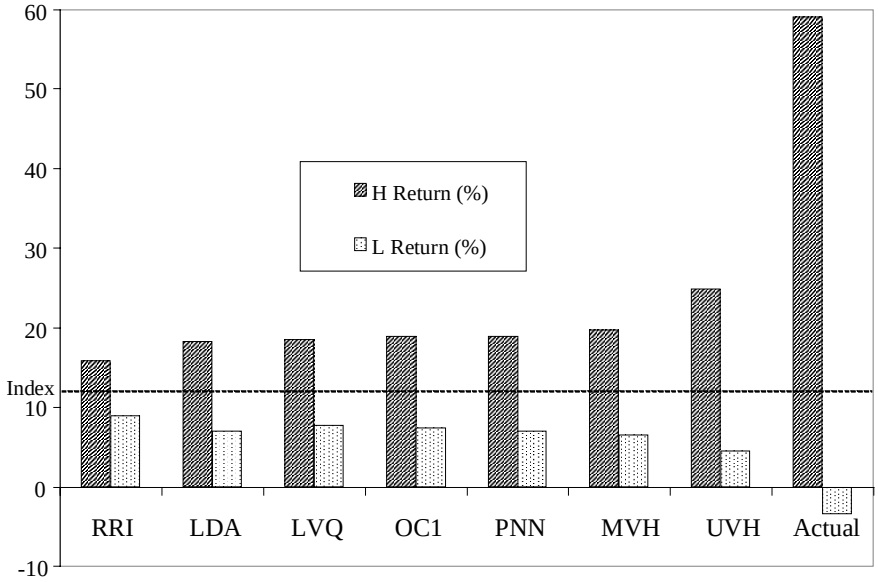

Figure 6. Returns to Predicted High and Low Portfolios