Available online at www.sciencedirect.com

SCIENCE @ DIRECT®

ELSEVIER

Computers & Operations Research 32 (2005) 2543–2559

computers & operations research

www.elsevier.com/locate/dsw

Neural network ensemble strategies for financial decision applications

David West*, Scott Dellana, Jingxia Qian

Department of Decision Sciences, College of Business Administration, East Carolina University, Greenville, NC 27836, USA Voice 252-328-6370

Accepted 29 March 2004

Available online 25 May 2004

Abstract

Considerable research effort has been expended to identify more accurate models for decision support systems in financial decision domains including credit scoring and bankruptcy prediction. The focus of this earlier work has been to identify the "single best" prediction model from a collection that includes simple parametric models, nonparametric models that directly estimate data densities, and nonlinear pattern recognition models such as neural networks. Recent theories suggest this work may be misguided in that ensembles of predictors provide more accurate generalization than the reliance on a single model. This paper investigates three recent ensemble strategies: crossvalidation, bagging, and boosting. We employ the multilayer perceptron neural network as a base classifier. The generalization ability of the neural network ensemble is found to be superior to the single best model for three real world financial decision applications.

Keywords: Credit scoring; Bankruptcy prediction; Ensembles; Bagging; Boosting

1. Introduction

Financial credit is an immense global industry. In the United States alone the annual transactions of Visa, Mastercard, Discover, and American Express credit cards totaled $1.2 trillion from over 500 million cards in circulation. The outstanding level of consumer debt in the U.S. totals about $1.5 trillion, with high interest credit card loans comprising $568.4 billion of that total. More than 4% of credit card loans are delinquent and placed for collection every year. U.S. bankruptcy filings for the

* Corresponding author. Tel.: +1-252-3286370; fax: +1-919-3284092.
E-mail addresses: westd@mail.ecu.edu (D. West), dellanas@mail.ecu.edu (S. Dellana).

0305-0548/$ - see front matter © 2004 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cor.2004.03.017

year 2002–2003 set a record level, totaling 1,650,279, which includes 37,182 business bankruptcy filings.

There is a clear need for accurate decision support for both the credit granting decision and the monitoring of the ongoing health of credit customers. An improvement in accuracy of even a fraction of a percent translates into significant future savings for the credit industry.

Traditional methods of financial decision support include scorecards for consumer credit [1–5] and discriminant models for assessing corporate financial health [6,4]. Both are essentially multivariate linear models that output a probability that the client will repay debt as agreed. Recent research interest has focused on more complex nonlinear models, particularly neural networks, to increase the credit decision accuracy [2,6–19]. The reader is referred to Smith and Gupta [20] for a recent survey of the application of neural networks in a diverse range of operations research problems that include financial forecasting and creditworthiness.

The focus of prior research has been to identify the "single best" model that is most accurate for a given financial decision application. This reliance on a single model may be misguided. Recent studies of ensembles (or committees) of predictors have demonstrated the potential to reduce the generalization error of a single model from 5% to 70% [21,22]. Three major strategies have been advanced for forming ensembles of predictors. The simplest is the crossvalidation (CV) neural network ensemble where all ensemble members are trained with the same data [23,16]. The second and third strategies create perturbed versions of the training set so that ensemble members learn from different variants of the original training data. Bagging ensembles create a unique training set for each ensemble member by sampling with replacement over a uniform probability distribution on the original data [21,24,25]. This creates training sets where some observations are replicated and others may be missing. Boosting is also a re-sampling strategy, with a probability distribution that is dependent on the misclassification rate for each observation [26,16]. Boosting is an iterative algorithm where the probability of the misclassified observations is increased and the corresponding probability of correctly classified observations is decreased over time. As boosting progresses, the composition of the training sets becomes increasingly dominated by hard-to-classify examples. The purpose of this research is to investigate the accuracy of ensembles of neural networks formed from these three strategies for credit granting and bankruptcy decision applications.

In the next section of this paper we review the recent theory and application of ensembles, with particular attention given to neural networks. Specific research questions are defined in this section. The research methodology is described in Section 3, and in Section 4 the comparison of generalization errors for the neural network ensemble strategies is discussed. We conclude in Section 5 with guidelines for implementing neural network ensembles for financial decision applications.

2. Ensemble strategies

The basic concept of the ensemble method is that diverse perspectives on different aspects of a problem can be combined to produce a high quality decision. For example, O'Leary [27] investigated human performance in the task of knowledge acquisition of probability estimates. He compared the relative performance of individuals versus groups of "multiple experts" (i.e., ensembles). His results suggest that knowledge acquisition from groups provided more correct probability orderings than the orderings from individuals acting alone. This finding is consistent with earlier research cited by O'Leary concluding that individuals exhibit fallacies in their probability reasoning [27]. Similar research has been conducted with machine learning algorithms for decision support systems [28]. Zhou and Lopresti [29] found that a consensus vote of multiple machine learning models trained by repeated sampling always yields a net improvement in recognition accuracy for common distributions of interest. These findings suggest that ensembles or collections of learning models provide more accurate problem generalization than the selection of a "single best" model determined from cross validation tests [21,22,30–32].

Hansen and Salamon provide some of the first research results to demonstrate that the generalization error of a neural network can be significantly reduced by using an ensemble of similar networks, all trained with the same data [33]. They referred to this strategy as a crossvalidation (CV) ensemble [34]. An explanation offered by Hansen and Salamon for the performance advantage of the CV ensemble is that the multitude of local minima encountered in the training of individual neural networks results in errors occurring in different regions of input space. The collective decision of the ensemble is, therefore, less likely to be in error than the decision made by any of the individual networks [33]. There are some recent studies of the use of genetic algorithms to train a population of neural networks that have some similarities to CV ensemble strategies. For example, Pendharkar [35] investigated hybrid approaches that include both evolutionary and neural training algorithms. Pendharkar used a population of 100 CV neural networks and a genetic algorithm to evolve a pool of neural network candidates with near optimal weights.

Another prominent ensemble strategy is "bootstrap aggregating", or "bagging", championed by Breiman [21,22] and used by Zhang [30,31]. A bagging ensemble is formed by perturbing the training data, creating a unique training set for each ensemble member by sampling with replacement over a uniform probability distribution on the original data [21,24,25]. This creates training sets where some observations are replicated and others may be missing. A key distinction of bagging is that each of the "expert models" is trained under differing conditions, and an algorithm is applied to the model outputs (usually majority vote) to produce a single "expert decision". Breiman [21] investigated bootstrap replicates to create diverse learning sets for classification trees and tested them on both real and simulated data sets. Breiman reported a reduction in test set misclassifications (comparing the "bagging estimator" to the "single best" estimator) ranging from 6% to 77% and concluded that a vital element for the success of bagging is the instability of the estimators. If perturbing the learning set can cause a significant change in the predictor constructed, then bagging can improve the generalization accuracy. For classification problems, Breiman demonstrates that if a model's prediction is "order-correct" for most inputs, then an aggregated predictor or bagging model can be transformed into a nearly optimal predictor [21]. Using an approach similar to Breiman's work, Zhang [30,31] explores bagging an ensemble of thirty multiplayer perceptron neural network models on learning sets created by bootstrap replicates and concludes that the bagging estimator is more accurate and more robust than the "single best" neural network.

AdaBoost, or "adaptive boosting", is another ensemble strategy that uses perturbation in an attempt to improve the performance of the learning algorithm [36]. In this paper we will refer to this algorithm as simply "boosting", although the reader should be aware that there are a number of variants that have been proposed. Boosting is a re-sampling strategy, with a probability distribution that is dependent on the misclassification rate for each observation [26,16]. Boosting employs an iterative algorithm that constructs an ensemble by sequentially training each ensemble member with unique training sets that increase the prominence of certain hard-to-learn examples misclassified by earlier ensemble members. Boosting maintains a probability distribution, $D_t(t)$, over the original data available for training. In each iteration, a classifier(t) is trained by sampling with replacement from this distribution. After training and testing, the probability of incorrectly classified training examples is increased and the probability of correctly classified examples is decreased. The ensemble decision is obtained by a weighted vote of all ensemble members [36]. Schwenk and Bengio applied AdaBoost methods to neural network ensembles and report that boosting can significantly improve neural network classifiers [37]. They conclude that boosting is always superior to bagging, although the differences are not always significant.

Ensemble strategies have been investigated in several application domains. For example, Hu and Tsoukalas report that ensembles of multilayer perceptron (MLP) neural networks reduce the error in predicting the relative importance of situational and demographic factors on consumer choice [38]. Sohn and Lee employ ensembles of neural networks to improve the classification accuracy of road traffic accident severity [39]. They tested both bagging and boosting ensembles, and report a reduction in generalization error for the bagging neural network ensemble of 6.3% relative to the single neural network model [39]. Hayashi and Setiono report increased accuracy diagnosing hepatobiliary disorders from ensembles of 30 MLP neural networks [40]. They employed CV ensembles (without the data perturbation methods of bagging or boosting) with a relatively small training set size of 373 examples [40]. Zhou et al. employed ensembles of neural networks to identify lung cancer cells from needle biopsies [32]. The authors report high overall accuracy and a low rate of false negatives. Zhang aggregated 30 MLP neural networks to estimate polymer reactor quality [30], while Cunningham et al. report improved diagnostic prediction for medical diagnostic decision support systems that aggregate neural network models [41]. Zhilkin and Somorjai explore bagging ensembles by using combinations of linear and quadratic discriminant analysis, logistic regression, and MLP neural networks to classify brain spectra by magnetic resonance measurement [42]. They report that the bootstrap ensembles are more accurate in this application than any "single best" model.

The authors are not aware of any existing research on the application of ensemble strategies to financial decision applications. There is a need for more systematic study of the properties of neural network ensembles and the relative performance of these ensembles in financial classification applications. The contribution of this paper is to investigate the potential for reducing the generalization error of financial decision applications by forming neural network ensembles with each of the three main strategies (CV, bagging, and boosting). We conduct a controlled experiment capable of producing statistically significant conclusions about the relative performance of the three ensemble strategies and the strategy of reliance on a single model. Our focus is on two-group classification in financial credit scoring and bankruptcy prediction problems where the spatial data structure is characterized by two naturally occurring clusters. To further these purposes, we pose the following research questions.

1. Can a neural network ensemble be expected to universally produce a more accurate financial application decision than the strategy of relying on a single neural network model for a decision, or are there some conditions where the single model is preferred?
2. Are the neural network ensemble strategies that perturb the training set more accurate than the simple CV ensemble?
3. Does the complexity of the classification problem influence the relative performance of the neural network ensemble strategy?

For this research, we consider an ensemble to be a collection of a finite number of neural networks that are individually trained and whose member predictions are combined to create an aggregate decision. Specific algorithms for bagging and boosting are given in Section 3.

3. Research methodology

In this research, all neural network ensembles are constructed with 100 members. This quantity is based on findings that accurate boosting ensembles require a relatively large ensemble membership [22]. We focus exclusively on ensembles formed with the popular multilayer perceptron (MLP) neural network trained with the backpropagation algorithm. All MLP networks have a single hidden layer; the number of hidden neurons is determined for each of the 100 members by randomly sampling with replacement from the integer interval [2, 3,... number of network inputs]. A total of $i = 100$ experimental iterations are conducted for each data set by randomly shuffling the data rows and partitioning the data set into a learning set $\boldsymbol{L}_i$ with 70% of the examples, a validation set $\boldsymbol{V}_i$ with 15%, and an independent holdout test set $\boldsymbol{T}_i$ with 15%. These iterations allow us to contrast generalization errors between the three ensemble strategies (CV, bagging, and boosting) and the single best model and to detect relatively small differences in performance.

3.1. Description of data sets

The three real world financial data sets summarized in Table 1 are used to test the predictive accuracy of the ensemble strategies investigated. Dr. Hans Hofmann of the University of Hamburg contributed the German credit scoring data. It consists of 700 examples of creditworthy applicants and 300 examples where credit should not be extended. For each applicant, 24 variables (3 continuous and 21 categorical of which 9 are binary), describe credit history, account balances, loan purpose, loan amount, employment status, personal information, age, housing, and job. This data set has a relatively high noise/signal ratio, which confounds the task of learning a classification function. The Australian credit scoring data [43] is similar but more balanced with 307 and 383 examples of each outcome. The data set contains a mixture of six continuous and eight categorical variables. To protect the confidentiality of this data, attribute names and values have been changed to symbolic data. The reader is cautioned that neither data set contains credit bureau information, which is usually available to the credit granting institution. The German credit data also contains some information like gender, marital status, and nationality that cannot legally be used in the U.S. The bankruptcy data set was constructed by the authors from Standard and Poor's Compustat financial files. This data set consists of five key financial ratios from Altman's research [6]. These ratios, constructed from financial statement information two years prior to bankruptcy include: working capital/total assets, retained earnings/total assets, earnings before interest and taxes/total assets, market value of equity/book value total liability, and sales/total assets. There are a total of 329 observations in the data set with 93 bankrupt companies and 236 healthy companies. We use labels of -1 (bad credit/bankrupt) and $+1$ (good credit/healthy) for all three data sets.

Table 1
Data set characteristics

	Australian credit	German credit	Bankruptcy data
Number of Examples	690	1000	329
Proportion of Bad Credit	0.445	0.300	0.283
Number of Predictors	14	24	55
Binary Predictors	4	9	0
Mean Correlation of Predictors	0.102	0.085	0.091
Skewness	1.97	1.70	2.05
Kurtosiss	12.55	7.79	10.12
Noise/Signal	19.36	79.38	14.67

3.2. Experimental design

A data partitioning strategy is employed in this research that follows the general spirit of earlier work [21,22,31,32]. A total of $i = 100$ different experimental iterations are created for each data set by randomly shuffling the data rows and partitioning the data into a learning set $\boldsymbol{L}_i$ with 70% of the observations, a validation set $\boldsymbol{V}_i$ with 15%, and an independent test set $\boldsymbol{T}_i$ with 15%. For each of the 100 iterations, the neural network model (with randomly determined hidden layer architecture) is trained with $\boldsymbol{L}_i$ and the generalization error on the independent test set $\boldsymbol{T}_i$ is assessed. The validation set $\boldsymbol{V}_i$ is used during this process to implement early stopping and avoid model over-fitting.

The estimate of the generalization error of the single best model is determined for each iteration by training all potential MLP models with $\boldsymbol{L}_i$ using $\boldsymbol{V}_i$ to implement early stopping. Recall that all potential MLP models include hidden layer neurons varying from a minimum of two to a maximum that equals the number of network inputs. The hyperbolic tangent activation function is used in the neural network to map to outputs of ±1. The single most accurate model is identified from the set of potential models based on a minimum validation error. The generalization error of that model is then measured on the independent test set, $\boldsymbol{T}_i$. An estimate of the mean generalization error for the single model is then determined by averaging the generalization error for the 100 iterations.

A corresponding estimate of the generalization error for the three ensemble strategies is obtained using the same data partitions and neural network architectures employed for the single best model. The CV ensemble does not require any perturbation of the learning set $\boldsymbol{L}_i$. Each MLP architecture is trained on $\boldsymbol{L}_i$ for $i = 1, \ldots, 100$, with $\boldsymbol{V}_i$ again used to implement early stopping of training and avoid overfitting. The decision of each MLP ensemble member ($+1$ or -1) is then recorded for each instance of the test set $\boldsymbol{T}_i$. An aggregate decision for the CV ensemble is simply a majority vote of the individual ensemble members.

In this research, bagging ensembles are constructed by forming 100 bootstrap replicates training sets, $\boldsymbol{L}_{iB}$, for each of the 100 iterations. Bootstrap replicates are formed by sampling with replacement from the original training set partition $\boldsymbol{L}_i$. An implication of the bootstrap process is that some of the original training set observations will be missing in $\boldsymbol{L}_{iB}$, the bagging training set, while other observations may be replicated several times. This provides diversity in the training set, which may potentially decrease the ensemble generalization error. The specific algorithm for creating bagging ensembles follows.

Algorithm for bagging ensemble

Given: training set of size n and base classification algorithm $C_t(\mathbf{x})$

```
Step 1.   Input sequence of training samples $(x_1, y_1), \ldots (x_n, y_n)$ with labels $y \in Y = (-1, 1)$
Step 2.   Initialize probability for each example in learning set $D_1(i) = 1/n$ and set $t = 1$
Step 3.   Loop while $t < B{=}100$ ensemble members
              a. Form training set of size n by sampling with replacement from distribution $D_t$
              b. Get hypothesis $ht : X \to Y$
              c. Set $t = t + 1$
          End of loop
Step 4.   Output the final ensemble hypothesis
```

$$\mathbf{C}^*(x_i) = \boldsymbol{h}_{\text{final}}(\boldsymbol{x}_i) = \text{argmax} \sum_{t=1}^{B} \boldsymbol{I}(\boldsymbol{C}_t(\boldsymbol{x}) = \boldsymbol{y}).$$

A boosting ensemble is also constructed by perturbing the original training set. For boosting, the sampling process is controlled by a probability distribution, $D_t(i)$, maintained over the observations in $\boldsymbol{L}_i$. For the construction of the first ensemble member, $D_t(i)$ is a uniform distribution. At each iteration a training set is generated by sampling with replacement from $D_t(i)$ (Step 3a), the neural network is trained and tested (Step 3b), and a weighted error is calculated from the sum of the probabilities of misclassified observations (Step 3c). An inflation factor β_t is calculated in Step 3e from the weighted error. In Step 3f, the probability of all misclassified observations is increased by multiplying the probability of each misclassified observation by the inflation factor β_t and then $D_t(i)$ is renormalized. If the weighted error is ever less than zero or greater than 0.5, the distribution $D_t(i)$ is reset to a uniform distribution (Step 3d). After 100 ensemble members have been constructed in this fashion, the ensemble decision is determined by weighting the decision of each ensemble member by $\log(\beta_t)$ (Step 4). The specific boosting algorithm used in this research follows.

Algorithm for AdaBoost ensemble

Given: training set of size n and base classification algorithm $C_t(\mathbf{x})$

```
Step 1.   Input sequence of training samples $(x_1, y_1), \ldots (x_n, y_n)$ with labels $y \in Y = (-1, 1)$
Step 2.   Initialize probability for each example in learning set $D_1(i) = 1/$n and set $t = 1$
Step 3.   Loop while $t < B{=}100$ ensemble members
              a. Form training set of size n by sampling with replacement from distribution $D_t$
              b. Get hypothesis ht : $X \to Y$
              c. Calculate the weighted error rate: $\varepsilon_t = \sum_{i:h_t(x_i) \neq y_i} \boldsymbol{D}_t(\boldsymbol{i})$
              d. If $\varepsilon_t \leqslant \mathbf{0}$ or $\varepsilon_t \geqslant \mathbf{0.5}$ then reinitialize $D_t(i) = 1/n$ and GOTO step 3a
              e. Calculate $\beta_t = (\mathbf{1} - \varepsilon_t)/\varepsilon_t$
              f. Update probability distribution: $\boldsymbol{D}_{t+1}(\boldsymbol{i}) = \frac{\boldsymbol{D}_t(\boldsymbol{i})\beta_t^{I(h_t(x_i \neq y_i))}}{\boldsymbol{Z}_t}$ where $Z_t$ is a normalization
                 constant
              g. Set $t = t + 1$
          End of loop
Step 4.   Output the final ensemble hypothesis $\boldsymbol{C}^*(\boldsymbol{x}_i) = \boldsymbol{h}_{\text{final}}(\boldsymbol{x}_i) = \sum \log(\beta_t)$
```

Table 2
Australian credit ensemble error (100 experimental replications)

	Average 100 members	Average 50–100 members	Minimum 50–100 members	Maximum 50–100 members
CV ensemble	0.131262	0.131315	0.130097	0.132718
Bagging ensemble	0.127961	0.128252	0.125243	0.132524
Boosting ensemble	0.14767	0.149214	0.134951	0.163301
Single model	0.132271	NA	NA	NA

Table 3
German credit ensemble error (100 experimental replications)

	Average 100 members	Average 50–100 members	Minimum 50–100 members	Maximum 50–100 members
CV ensemble	0.241867	0.236933	0.241758	0.246867
Bagging ensemble	0.251333	0.251758	0.243467	0.259400
Boosting ensemble	0.254733	0.256442	0.244400	0.268800
Single model	0.253117	NA	NA	NA

Table 4
Bankruptcy ensemble error (100 experimental replications)

	Average 100 members	Average 50–100 members	Minimum 50–100 members	Maximum 50–100 members
CV ensemble	0.129184	0.129425	0.127959	0.130816
Bagging ensemble	0.126327	0.125937	0.122041	0.130408
Boosting ensemble	0.127551	0.128293	0.116531	0.14
Single model	0.131429	NA	NA	NA

4. Ensemble generalization results

The generalization errors for each data set investigated are reported in Tables 2–4. The single most accurate model result is determined by training each potential MLP model and identifying the most accurate model based on its classification accuracy on the validation examples. In this research, potential models include hidden layer neurons ranging from 2 to the number of input features in the data set. The generalization error of the most accurate model is then assessed by classifying the independent holdout sample. We report the average generalization error for an ensemble formed from 100 MLP members and an average error across all ensembles with 50 or more members. We also include an average of the minimum and maximum errors for ensembles of 50 members or more. All averages are calculated across the 100 iterations conducted for this research. A figure depicting the generalization error as a function of the number of ensemble members is also included for each data set. The results are reported by data set in the next three subsections.

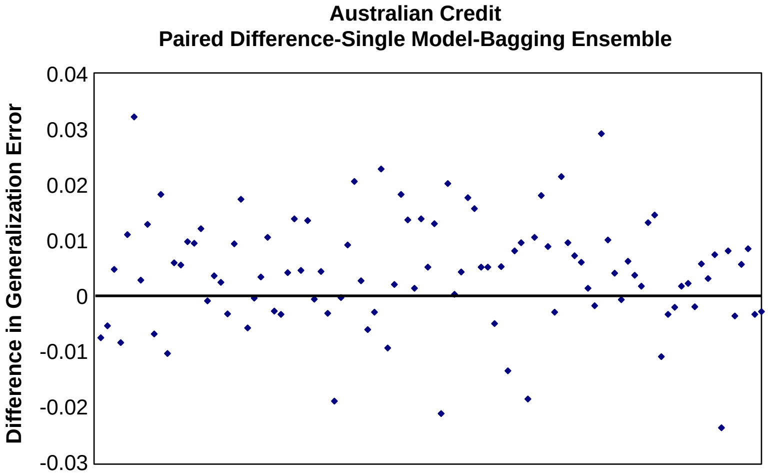

Fig. 1. Australian credit paired difference-single model-bagging ensemble.

4.1. Australian credit

The generalization errors for the Australian credit data are reported in Table 2. For this data set, the bagging ensemble has the lowest average error of 0.127961 for ensembles of 100 members. This compares to errors of 0.131262 for the CV ensemble and 0.14767 for the boosting ensemble. The average generalization error of the 100-member bagging ensemble is a reduction of 3.26% from the single best model error of 0.132271. In an earlier study of this data set, West [18] reports a single model MLP error of 0.1416 using a 10-fold crossvalidation partitioning scheme. A paired t-test on the differences in generalization errors between the bagging ensemble and the single best model for the 100 experimental replications concludes that the bagging ensemble achieves a statistically significant reduction in generalization error: $p = 0.0000$. A scatter plot of the paired differences is presented in Fig. 1. The average maximum and minimum errors obtained confirm that the boosting ensemble is more erratic and unstable than the CV or bagging ensemble (see Table 2). The average minimum error for boosting is 0.134951 and the average maximum error is 0.163301. This compares to ranges of 0.130097–0.0132718 for CV ensembles and 0.125243–0.132524 for bagging ensembles. This is confirmed by prior research findings that boosting is a relatively unstable ensemble strategy [24].

The average generalization error of the ensembles formed for the Australian credit data are shown graphically in Fig. 2 as a function of the number of ensemble members. The solid horizontal line represents the error determined for the single most accurate model. Note that the number of ensemble members has no meaning for the single model case. Fig. 2 suggests that boosting is not an effective ensemble strategy for this data set for any ensemble size. The fact that boosting is unpredictable, working well in some applications and not in others, has been reported in the literature previously [24]. It is also evident from Fig. 2 that the number of ensemble members has a pronounced effect on the boosting ensemble error; the generalization error continuously declines with increasing membership. This is consistent with Breiman's experience that 100 or more members may be required for

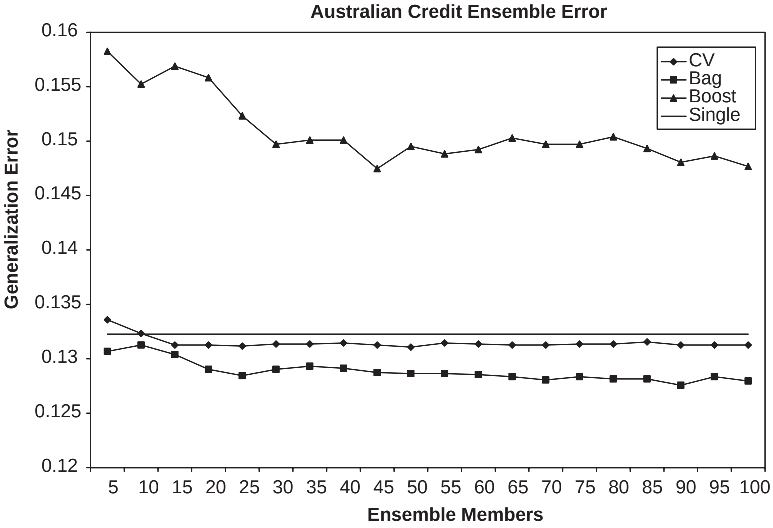

Fig. 2. Australian credit ensemble error.

boosting ensembles with CART as a baseline classifier [22]. By contrast, the generalization errors of the CV and bagging ensembles remain fairly constant above 30 members.

4.2. German credit

The average generalization errors of ensembles investigated for the German credit data are reported in Table 3. For this data, the CV ensemble has the lowest average generalization error of 0.241867, compared to 0.251333 for bagging ensembles and 0.254733 for boosting ensembles. The CV ensemble achieves a 4.44% error reduction from the single model result of 0.253117. West reported an MLP single model error of 0.2672 for 10-fold crossvalidation [18]. A paired t-test contrasting CV ensembles and single model errors concludes that this is a statistically significant reduction: $p = 0.0000$. Fig. 3 is a scatter plot of the differences in error between the single best model and the CV ensemble for the German credit data. The average of the maximum and minimum errors reported in Table 3 again illuminate that the boosting ensemble is more unstable, creating a higher range of errors.

The generalization error of German credit ensembles is presented in Fig. 4 as a function of the number of ensemble members. The performance of the boosting ensemble parallels the Australian credit experience in that the boosting error is greater than the single model error at all ensemble sizes investigated. Again we notice the pronounced effect that the number of ensemble members has on the boosting generalization error. In this application, neither of the two data perturbation ensemble strategies improves on the error performance of the single model, although the bagging ensemble with 60 or more members has a slightly lower error. The CV ensemble however, is significantly more accurate than the single model with ensembles formed from 5 to 100 members.

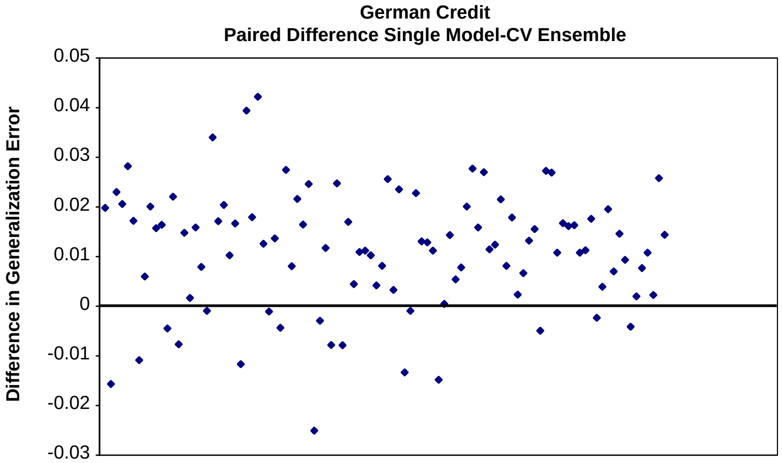

Fig. 3. German credit paired difference single model-CV ensemble.

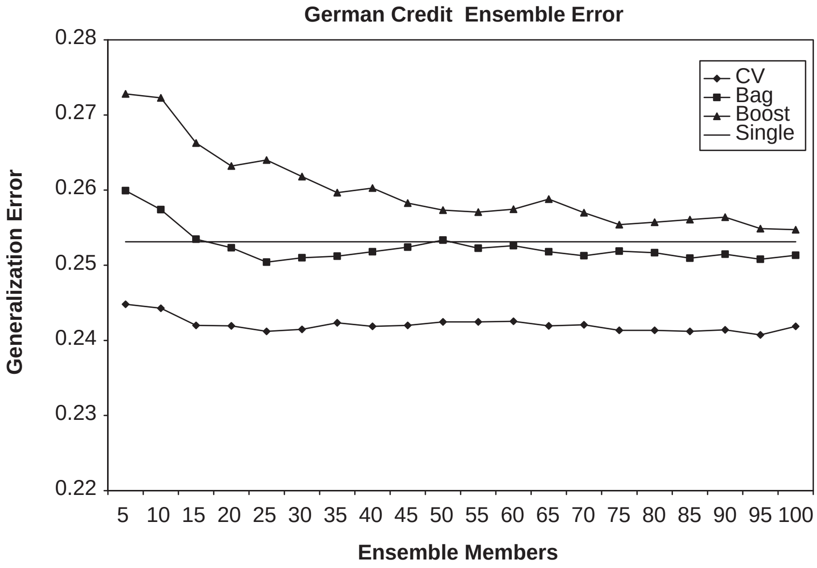

Fig. 4. German credit ensemble error.

4.3. Bankruptcy data

Table 4 summarizes the average generalization error of each ensemble strategy for the bankruptcy data set. The bagging ensemble with 100 members has the lowest generalization error of 0.126327 compared to 0.127551 for boosting and 0.129184 for the CV ensemble. The bagging ensemble error is a 3.88% error reduction from the single model result of 0.131429. A paired *t*-test of this difference between the bagging ensemble and the single model concludes a statistically significant reduction:

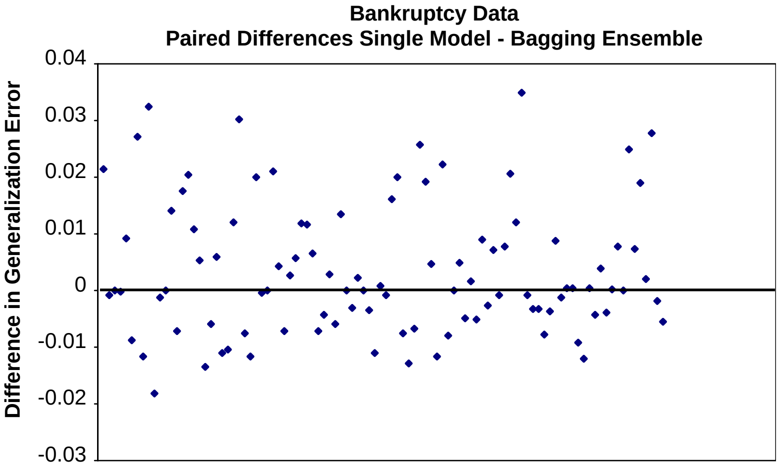

Fig. 5. Bankruptcy data paired differences single model—bagging ensemble.

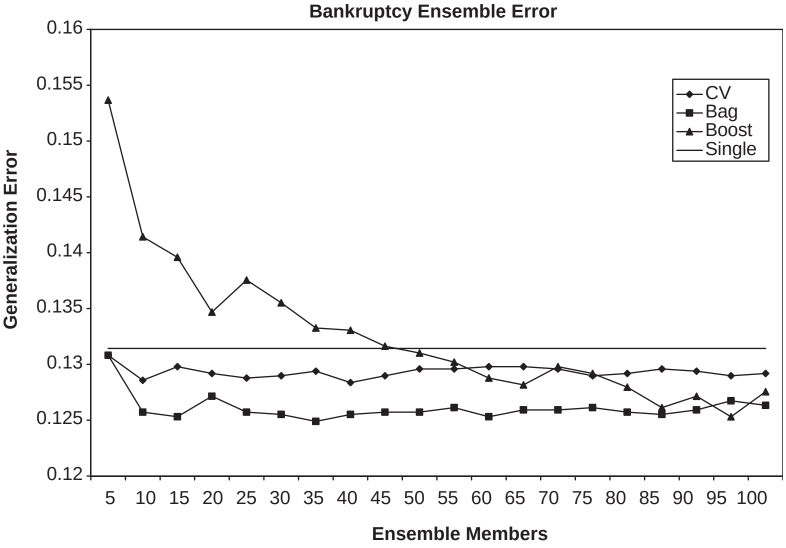

Fig. 6. Bankruptcy ensemble error.

$p = 0.04$. The average maximum and minimum errors for the ensemble strategies again show more instability for the boosting ensemble, although in this application the boosting error is competitive with the most accurate ensembles. A scatter plot of differences is presented in Fig. 5.

The ensemble errors for the bankruptcy data are shown in Fig. 6 as a function of the number of ensemble members. It is evident from Fig. 6 that the boosting ensemble error shows a marked decline as the number of ensemble members increases from 5 to 100 members and that the boosting ensemble with 95 members has the lowest generalization error of any alternative investigated. Unlike

Table 5
Difference in mean generalization error (Boosting versus all others)

Boosting:	CV	Bagging	Single
Difference	0.00922 (5.2%)	0.00811 (4.7%)	0.00491 (2.8%)
p	0.0000	0.0000	0.0002

Table 6
Difference in mean generalization error (Single best versus CV and Bagging)

Single:	CV	Bagging
Difference	0.0043 (2.5%)	0.0034 (1.9%)
p	0.0000	0.0000

the two credit data sets, boosting works well in this application, providing a large ensemble size is used. The other data perturbation strategy, bagging, also achieves a low generalization error on this data set, comparable to the boosting error. Both bagging and boosting errors are lower that the CV ensemble error for ensemble sizes of 80 or more members.

4.4. Consolidated results for three data sets

In this subsection we analyze the consolidated performance of the three ensemble strategies investigated by contrasting paired differences in mean generalization error across all three data sets. We first compare the consolidated result from the boosting ensemble to the other ensemble strategies and to the single model as shown in Table 5. The boosting ensemble has a mean error 2.8% higher than the single model, 4.7% higher than the bagging ensemble, and 5.3% higher than the CV ensemble. All three differences are statistically significant as evidenced from the p values of the paired t-test given in the second row of Table 5. An explanation for the relatively poor performance of the boosting ensemble is likely the presence of noise, outliers, and mislabeled learning examples in the two credit data sets as reported by West [19]. Dietterich has documented the detrimental effects of noise on the boosting algorithm [24]. An active area of research today is the design of new boosting algorithms that are more robust in the presence of noise.

The consolidated generalization errors of the other two ensemble strategies (CV and bagging) are compared to the single best model results in Table 6. The CV ensemble achieves an error that is 2.5% less than the single model across all three data sets. The consolidated generalization error of the bagging model is 1.9% less than the single model. The reduction in error achieved by these two ensemble strategies is statistically significant for a paired t-test analysis: $p < 0.0000$. Unfortunately, the choice of a specific ensemble strategy for a particular application remains ambiguous. The CV ensemble is most accurate for the German credit data, while the bagging ensemble is most accurate for the Australian credit and bankruptcy data. We also note that boosting was competitive with bagging for the bankruptcy data.

We now apply the weight of the findings for the individual and consolidated data sets to the specific research questions that are the motivation for this research.

Question 1 asks, "*Can a neural network ensemble be expected to universally produce more accurate decision than the strategy of relying on a single model for a decision, or are there some conditions where a single model is preferred?*" We conclude that an ensemble strategy produces more accurate generalization than the strategy of relying on a single model for a decision. An ensemble strategy is significantly more accurate for each of the three data sets investigated, and both CV and bagging ensembles achieve statistically significant reductions in error when considering results aggregated across all three data sets.

Questions 2 asks, "*Are the ensemble strategies that perturb the training set more accurate than the simple CV ensemble?*" We find no evidence of the effectiveness of ensemble strategies that perturb the data. A comparison of the consolidated error difference for the CV ensemble (non-perturbing) and the bagging ensemble (perturbing) across all 3 data sets produces an error difference of 0.0011. The check of statistical significance suggests there is no difference in error performance gained by perturbing the data ($p = 0.16$).

Question 3 asks, "*Does the complexity of the classification problem influence the relative performance of the neural network ensemble strategy?*" While experience with just three data sets is not sufficient for strong conclusions, we observe that the boosting ensemble strategy is effective only for the low noise bankruptcy data. The bagging ensemble tolerates noise better than boosting, and is effective for the low and medium noise data. Neither of the perturbation strategies is competitive with the CV ensemble for the relatively high noise German credit data.

4.5. Sensitivity and specificity analysis

This research focuses not on the implementation of a financial decision system, but on understanding some general properties of neural network ensembles. For this type of research, it is fairly common to measure misclassification instances and not misclassification costs. While we consider the pursuit of different economic cost scenarios to be a diversion from the basic purpose of this research, we acknowledge that the costs of misclassification do vary considerably by classification group and must be considered during the implementation stage. We therefore illustrate one method of tuning the ensemble to specific decision economics. The practitioner can modify the ensemble voting threshold to implement specific costs of misclassification. In the prior analysis, we used majority vote to determine the ensemble decision. If 51 or more ensemble members classify an example as belonging to group $+1$, the ensemble decision is to assign this instance to group $+1$. If 49 or fewer ensemble members classify the example as group $+1$, then the ensemble assigns the instance to group -1. In the event of a tie, the ensemble classifies the instance to the group with the highest prior probability.

Different levels of sensitivity (the probability that a good credit example will be classified into the good credit group) and specificity (the probability that a bad credit instance will be classified into the bad credit group) are obtained by varying the ensemble voting threshold from 0 to 100. The resulting receiver operating characteristic (ROC) curve for the bankruptcy data using a 100 member bagging ensemble is presented in Fig. 7 [44]. The arrow on this figure identifies the location of the majority vote threshold on the ROC curve. At this voting threshold of 50, the sensitivity of the ensemble is 91.6% and the specificity is 77.4%. The sensitivity and specificity vary over the voting

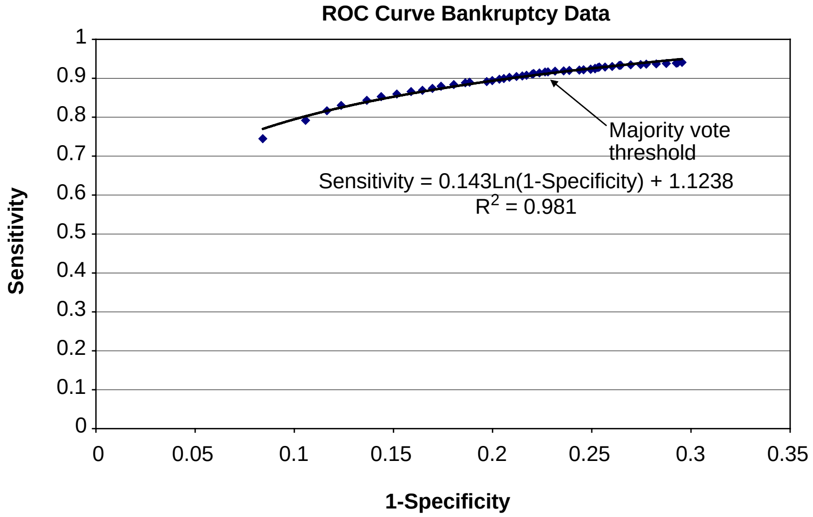

Fig. 7. ROC curve bankruptcy data.

threshold interval according to the following equation fitted to the data with an R^2 of 0.981.

$$\text{Sensitivity} = 0.143\text{Ln}(1 - \text{Specificity}) + 1.1238. \tag{1}$$

For decision implementations where the false positive rate is a major cost concern, the practitioner can increase the voting threshold for a +1 group assignment above the majority vote level of 50 and thereby increase the ensemble decision specificity. At the extreme threshold of 100, which requires unanimous consensus of all members to classify an instance as +1, the specificity for the bankruptcy ensemble is increased to 91.6%, while the sensitivity decreases to 74.5%. The reader is cautioned that prior group probabilities must also be considered in the determination of cost of misclassification. The reader is referred to West [38] for a more complete treatment of misclassification costs for the credit data sets.

5. Conclusions

Recent research has focused on the identification of higher capacity nonlinear models to improve the generalization accuracy of financial decision applications. A better approach than the reliance on a single high capacity model is to pursue ensemble strategies that combine the predictions of a collection of individual models. The results of this research confirm that ensembles of neural network predictors are more accurate and more robust that the "single best" MLP model based on experiments with three real world financial data sets. The ensemble strategies employed in this research reduced the generalization errors estimated for the single model case by 3.26% for the Australian credit, 4.44% for the German credit, and 3.88% for the bankruptcy data. All three differences are statistically significant reductions in generalization error. While an error reduction of 3–5% may seem modest, the reader must appreciate that the global credit industry has transactions exceeding $1 trillion annually. Based on an annual write off rate of 4%, a decision technology that is capable of reducing classification errors by 3% could potentially save the industry $1.2 billion annually.

It is more difficult to make a design recommendation for ensemble strategies that introduce diversity by intentionally perturbing the training set using bootstrap or boosting algorithms. Our aggregate analysis finds no significant difference in accuracy between the perturbation strategies and the simple CV ensemble. However, we do note that each of the three ensemble strategies investigated achieved a statistically significant reduction in error in at least one application. The CV ensemble was most accurate for the German credit data, an application characterized by high noise levels, a relatively large training set size, and a large number of feature variables. The bagging strategy was most effective for the Australian credit and the bankruptcy data set, both characterized by smaller training samples, fewer feature variables, and less noise. The boosting strategy was effective only for the bankruptcy data, the smallest data set with the fewest number of feature variables and the least amount of noise. While 25–30 ensemble members are adequate for the CV and bagging ensembles, we note as many as 100 members are necessary for the boosting ensemble.

References

[1] Brill J. The importance of credit scoring models in improving cash flow and collections. Business Credit 1998;100: 16–7.

[2] Henley WE. Statistical aspects of credit scoring. Dissertation, The Open University, Milton Keynes, UK, 1995.

[3] Mester LJ. What's the point of credit scoring? Business Review—Federal Reserve Bank of Philadelphia 1997: 3–16.

[4] Reichert AK, Cho CC, Wagner GM. An examination of the conceptual issues involved in developing credit-scoring models. Journal of Business and Economic Statistics 1983;1:101–14.

[5] Rosenberg E, Gleit A. Quantitative methods in credit management: a survey. Operations Research 1994;42:589–613.

[6] Altman EI, Marco G, Varetto F. Corporate distress diagnosis: comparisons using linear discriminant analysis and neural networks (the Italian experience). Journal of Banking and Finance 1994;18:505–29.

[7] Coats PK, Fant LF. A neural network approach to forecasting financial distress. Journal of Business Forecasting Winter 1991–92:9–12.

[8] Davis RH, Edelman DB, Gammerman AJ. Machine-learning algorithms for credit-card applications. IMA Journal of Mathematics Applied in Business & Industry 1992;4:43–51.

[9] Desai VS, Conway DG, Crook JN, Overstreet GA. Credit-scoring models in the credit-union environment using neural networks and genetic algorithms. IMA Journal of Mathematics Applied in Business & Industry 1997;8: 323–46.

[10] Desai VS, Crook JN, Overstreet GA. A comparison of neural networks and linear scoring models in the credit union environment. European Journal of Operational Research 1996;95:24–37.

[11] Efron B, Tibshirani RJ. An introduction to the bootstrap. New York: Chapman & Hall; 1993.

[12] Henley WE, Hand DJ. A k-nearest neighbor classifier for assessing consumer credit risk. Statistician 1996;44:77–95.

[13] Jensen HL. Using neural networks for credit scoring. Managerial Finance 1992;18:15–26.

[14] Lacher RC, Coats PK, Shanker CS, Fant LF. A neural network for classifying the financial health of a firm. European Journal of Operational Research 1995;85:53–65.

[15] Piramuthu S. Financial credit-risk evaluation with neural and neurofuzzy systems. European Journal of Operational Research 1999;112:310–21.

[16] Salchenberger LM, Cianr EM, Lash NA. Neural networks: a new tool for predicting thrift failures. Decision Sciences 1992;23:899–916.

[17] Tam FY, Kiang MY. Managerial applications of neural networks: the case of bank failure predictions. Management Science 1992;38:926–47.

[18] West D. Neural network credit scoring models. Computers & Operations Research 2000;27:1131–52.

[19] West D, Munchineuta C. Credit scoring using supervised and unsupervised neural networks. In: Smith K, Gupta J, editors. Neural networks in business: techniques and applications. Hershey, Pa.: Idea Publishing Group, 2002. p. 154–6.

[20] Smith KA, Gupta JND. Neural networks in business: techniques and applications for the operations researcher. Computers & Operations Research 2000;27:1023–44.

[21] Breiman L. Bagging predictors. Machine Learning 1996;24:123–40.

[22] Breiman L. Prediction games and arcing algorithms. Neural Computation 1999;11:1493–517.

[23] Frydman HE, Altman EI, Kao D. Introducing recursive partitioning for financial classification learning algorithms. Journal of Finance 1985;40:269–91.

[24] Dietterich TG. Experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting, and randomization. Machine Learning 2000;40:139–57.

[25] Efron B, Gong G. A leisurely look at the bootstrap, the jackknife, and cross-validation. The American Statistician 1983;37:36–48.

[26] Avnimelech R, Intrator N. Boosting regression estimators. Neural Computation 1999;11:499–520.

[27] O'Leary DE. Knowledge acquisition from multiple experts: an empirical study. Management Science 1998;44: 1049–58.

[28] LeBlanc M, Tibshirani R. Combining estimates in regression and classification. Journal of the American Statistical Society 1996;91:1641–50.

[29] Zhou J, Lopresti D. Improving classifier performance through repeated sampling. Pattern Recognition 1997;30: 1637–50.

[30] Zhang J. Inferential estimation of polymer quality using bootstrap aggregated neural networks. Neural Networks 1999;12:927–38.

[31] Zhang J. Developing robust non-linear models through boostrap aggregated neural networks. Neurocomputing 1999;25:93–113.

[32] Zhou ZH, Jiang Y, Yang YB, Chen SF. Lung cancer cell identification based on artificial neural network ensembles. Artificial Intelligence in Medicine 2002;24:25–36.

[33] Hansen LK, Salamon P. Neural network ensembles. IEEE Transactions on Pattern Analysis and Machine Intelligence 1990;12:993–1001.

[34] Stone M. Cross-validatory choice and assessment of statistical predictions. Journal of the Royal Statistical Society 1974;36:111–47.

[35] Pendharkar PC. An empirical study of design and testing of hybrid evolutionary-neural approach for classification. Omega-International Journal of Management Science 2001;29:361–74.

[36] Schapire RE. The strength of weak learnability. Machine Learning 1990;5:197–227.

[37] Schwenk H, Bengio Y. Boosting neural networks. Neural Computation 2000;12:1869–87.

[38] Hu MY, Tsoukalas C. Explaining consumer choice through neural networks: the stacked generalization approach. European Journal of Operational Research 2003;146:650–60.

[39] Sohn SY, Lee SH. Data fusion, ensemble and clustering to improve the classification accuracy for the severity of road traffic accidents in Korea. Safety Science 2003;41:1–14.

[40] Hayashi Y, Setiono R. Combining neural network predictions for medical diagnosis. Computers in Biology and Medicine 2002;32:237–46.

[41] Cunningham P, Carney J, Jacob S. Stability problems with artificial neural networks and the ensemble solution. Artificial Intelligence in Medicine 2000;20:217–25.

[42] Zhilkin PA, Somorjai RL. Application of several methods of classification fusion to magnetic resonance spectra. Connection Science 1996;8:427–42.

[43] Quinlan JR. Simplifying decision trees. International Journal of Man-Machine Studies 1987;27:221–34.

[44] Centor RM. Signal detectability: the use of ROC curves and their analyses. Medical Decision Making 1991;11: 102–6.