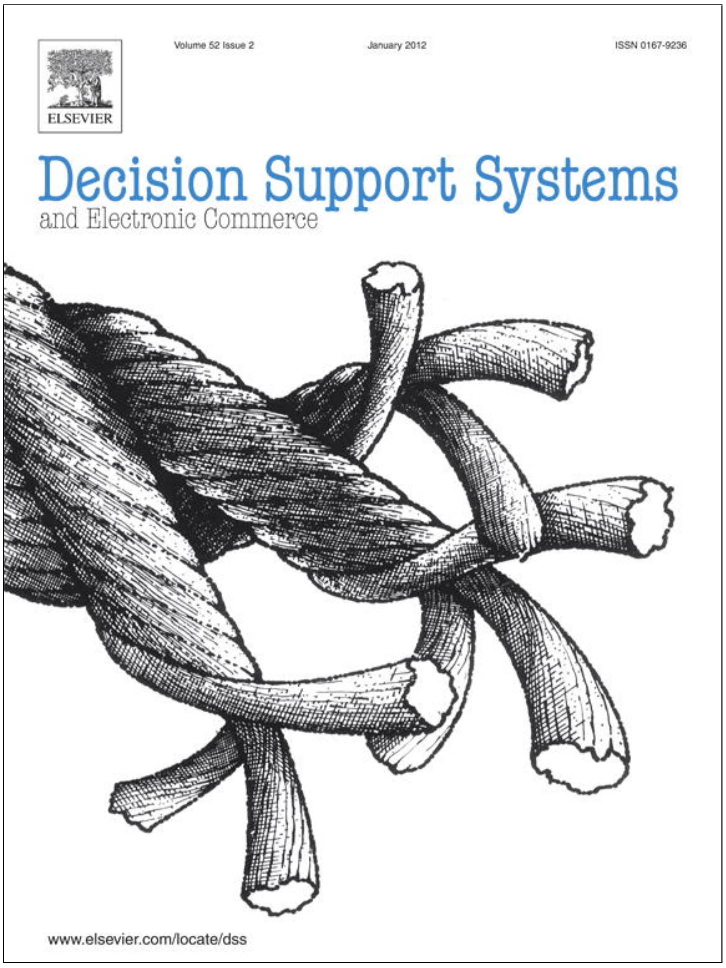
Volume 52 Issue 2
January 2012
ISSN 0167-9236
ELSEVIER
Decision Support Systems
and Electronic Commerce
www.elsevier.com/locate/dss

Decision Support Systems 52 (2012) 464–473

Contents lists available at SciVerse ScienceDirect

Decision Support Systems

journal homepage: www.elsevier.com/locate/dss

Comparative analysis of data mining methods for bankruptcy prediction

David L. Olson [a,*], Dursun Delen [b], Yanyan Meng [a]

[a] University of Nebraska, Management, 3300 Sheridan Court, Lincoln, NE 68506, United States
[b] State University, Management Science and Information Systems, United States

ARTICLE INFO

Article history:
Received 11 March 2011
Received in revised form 3 August 2011
Accepted 4 October 2011
Available online 13 October 2011

Keywords:
Bankruptcy prediction
Data mining
Neural networks
Decision trees
Support vector machines
Transparency
Transportability

ABSTRACT

A great deal of research has been devoted to prediction of bankruptcy, to include application of data mining. Neural networks, support vector machines, and other algorithms often fit data well, but because of lack of comprehensibility, they are considered black box technologies. Conversely, decision trees are more comprehensible by human users. However, sometimes far too many rules result in another form of incomprehensibility. The number of rules obtained from decision tree algorithms can be controlled to some degree through setting different minimum support levels. This study applies a variety of data mining tools to bankruptcy data, with the purpose of comparing accuracy and number of rules. For this data, decision trees were found to be relatively more accurate compared to neural networks and support vector machines, but there were more rule nodes than desired. Adjustment of minimum support yielded more tractable rule sets.

1. Introduction

Bankruptcy prediction has been a focus of study in business analytics because of the importance of accurate and timely strategic business decisions. Even though the accuracy of the prediction model is a very important criterion, understandability and transportability of the model are also important. The accurate prediction of bankruptcy has been a critical issue to shareholders, creditors, policy makers, and business managers.

There is a wealth of research that has been applied to this field [6,9,30,33,38,42], both in finance and in other fields [38]. Among the thousands of refereed journal articles, many recent studies have applied neural networks (NNs) [1,3,18,19,23,24,27,33,34,43,44,46,48]. Another popular approach is decision trees (DTs) [10,37,42,50]. Support vector machines (SVMs) have been proposed for smaller datasets with highly nonlinear relationships [12,15,21,35,40].

The vast majority of studies in this domain have focused on NNs, and how good they are compared to their statistical counterpart (i.e., logistic regression) at fitting data (fidelity [22]). However, neural network models are black boxes [4,51], lacking transparency (seeing what the model is doing, or comprehensibility) and transportability (being able to easily deploy the model into a decision support system for new cases). We argue that decision trees (DTs) can be as accurate, and provide transparency and transportability that NNs are often criticized for.

The paper is organized as follows. Section 2 reviews previous research on bankruptcy prediction based on data mining methods. Section 3 describes data mining methodologies. Section 4 discusses the data collected and Section 5 presents data analysis and prediction model building methods as well as the results obtained from different data mining techniques. Section 6 gives our conclusions.

2. Data mining model transparency

Model transparency relates to human ability to understand what the model consists of, leading ideally to the ability to apply it to new observations (which we might term transportability). If a model is transparent, it can be transported. Some models have consistently proven to be strong in their ability to fit data, such as neural network models, but to have low transparency or transportability. Neural networks by their nature involve highly complex sets of node connections and weights that can be obtained from software, but at a high cost in terms of transparency and transportability because there are so many nodes and weights. Conversely, logistic regression (or logit regression) have a form that can be understood and transported quite easily. Beta weights can be used to multiply times observation measures, yielding a score that can be used to classify new observations with relative ease. Support vector machines share the characteristics of transparency and transportability with neural network models. Decision tree models are highly transparent, yielding IF–THEN rules that are easier to comprehend and apply than even regression models.

Thus the issue of transparency primarily applies to neural network models. Önsei et al. [25] used neural network models to generate weights of 178 criteria which were then used in a model to classify country

* Corresponding author. Tel.: +1 402 472 4521; fax: +1 402 472 5855.
E-mail address: dolson3@unl.edu (D.L. Olson).

0167-9236/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.dss.2011.10.007

competitiveness. It has been recognized in the engineering field that neural network models need greater transparency [20]. There have been a number of applications [17,27,37] proposing a neurofuzzy framework to take advantage of neural network learning ability and rule-based transparency. Risser et al. [31] used neural networks to fit data, and jack-knife, bootstrap, and their own validation samples to obtain transparent models for evaluation of driver's license suspension. Yuan et al. [47] proposed a fuzzy neural network controller in the electronics field as a means to combine semantic transparency of rule-based fuzzy systems with the ability of neural networks to fit data. Chan et al. [8] used a similar approach to support vector regression models.

3. Data mining methodology

In a comparative analysis of multiple prediction models, it is a common practice to split the complete data set into training and testing sub sets, and compare and contrast the prediction models based on their accuracy on the test data set. In splitting the data into training and testing dataset one can choose to make a single split (e.g., half of the data for training and other half of the data for testing) or multiple splits, which is commonly referred to as k-fold cross validation. The idea behind k-fold cross validation is to minimize the bias associated with the random sampling of the training and holdout data samples. Specifically, in k-fold cross validation the complete data set is randomly split into k mutually exclusive subsets of approximately equal size. Each prediction model is trained and tested k times using exactly the same k data sets (i.e., folds). Each time, the model is trained on all but one folds and tested on the remaining single fold. The cross validation estimate of the overall accuracy of a model is calculated by averaging the k individual accuracy measures as shown in the following equation

$$OA = \frac{1}{k}\sum_{i=1}^{k} A_i$$

where OA stands for overall cross validation accuracy, k is the number of folds used, and A is the accuracy measure of each folds.

Since the cross-validation accuracy would depend on the random assignment of the individual cases into k distinct folds, a common practice is to stratify the folds themselves. In stratified k-fold cross validation, the folds are created in a way that they contain approximately the same proportion of predictor labels as the original dataset. Empirical studies showed that stratified cross validation tend to generate comparison results with lower bias and lower variance when compared to regular cross-validation [16]. In this study, to estimate the performance of predictors a stratified 10-fold cross validation approach is used. Empirical studies showed that 10 seem to be an "optimal" number of folds (that balances the time it takes to complete the test and the bias and variance associated with the validation process) [7,16]. The methodology followed in the study is depicted in Fig. 1.

3.1. Prediction methods

In this study, several popular classification methods (e.g., artificial neural networks, decision trees, support vector machines and logistic

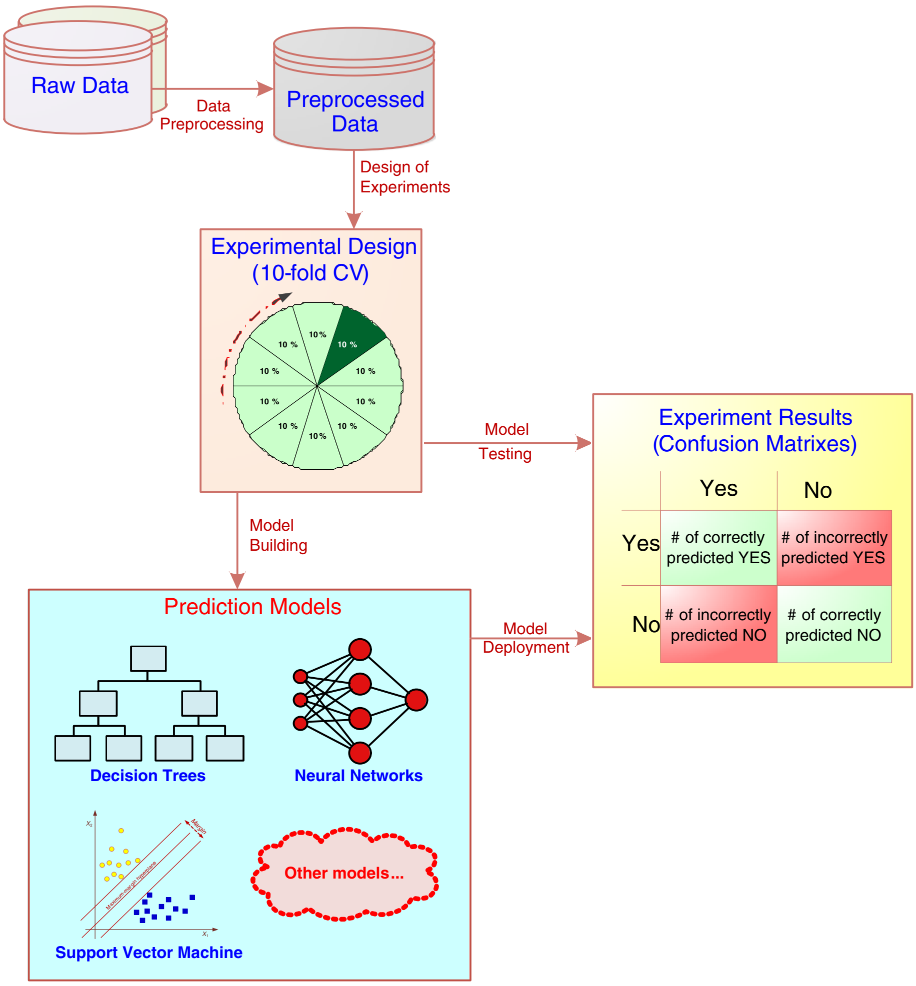

Fig. 1. A graphical depiction of the methodology followed in this study.

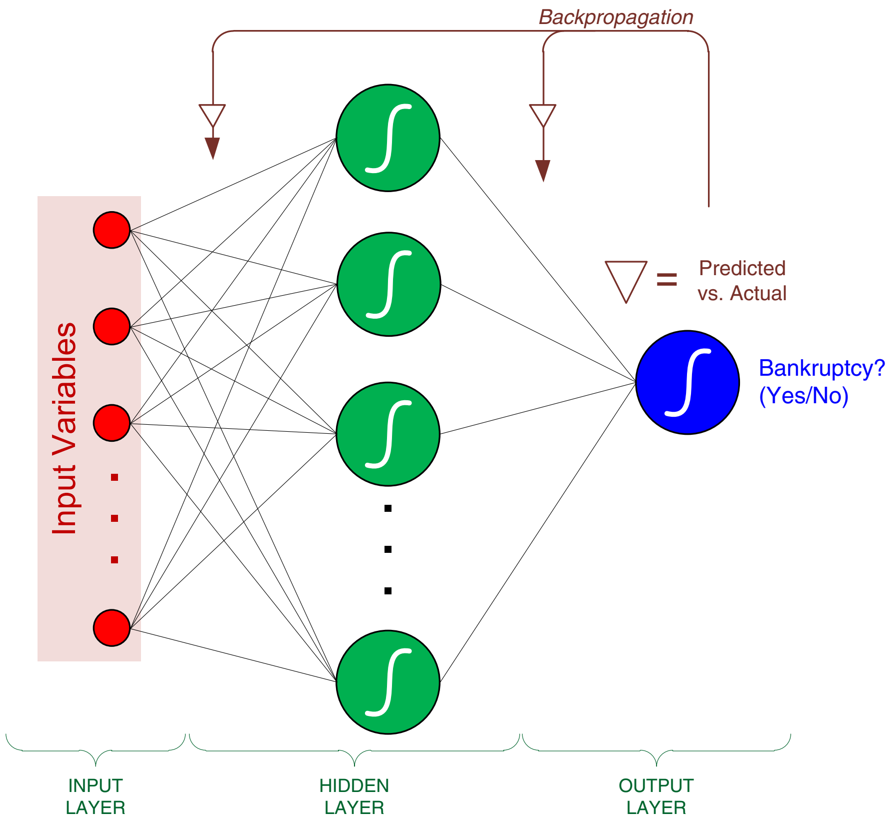

Fig. 2. The MLP architecture used in this study.

regression) are applied and compared to each other using predictive accuracy on the test data samples. What follows are brief descriptions of the prediction models used in this study:

Artificial neural networks (ANN) are biologically inspired analytical techniques, capable of modeling extremely complex non-linear functions [13]. In this study we used two popular neural network architectures, multi-layer perceptron (MLP) with a back-propagation and radial basis function (RBF). These two supervised learning algorithms are strong function approximators for prediction as well as classification type prediction problems. These two are arguably the most commonly used and well-studied ANN architectures. Even though they are comparable to each other in term of their prediction ability, Hornik et al. [14] empirically showed that given the right size and structure, MLP is capable of learning arbitrarily complex nonlinear functions to an arbitrary accuracy level. A pictorial representation of the ANN architecture used in this study is shown in Fig. 2.

Table 1
Attributes in bankruptcy data.

No	Short name	Long name
1	fyear	Data year — fiscal
2	cik	CIK number
3	at	Assets — total
4	bkvlps	Book value per share
5	invt	Inventories — total
6	Lt	Liabilities — total
7	rectr	Receivables — trade
8	cogs	Cost of goods sold
9	dvt	Dividends — total
10	ebit	Earnings before interest and taxes
11	gp	Gross profit (loss)
12	ni	Net income (loss)
13	oiadp	Operating income after depreciation
14	revt	Revenue — total
15	sale	Sales-turnover (net)
16	dvpsx_f	Dividends per share – ex-date – fiscal
17	mkvalt	Market value – total – fiscal
18	prch_f	Price high – annual – fiscal
19	bankruptcy	Bankruptcy (output variable)

Decision trees are powerful classification algorithms that are becoming increasingly more popular due to their intuitive explainability characteristics. Popular decision tree algorithms include Quinlan's [28,29] ID3, C4.5, C5, and Breiman et al.'s [5] CART (Classification and Regression Trees), Best First Decision Tree and AD Decision Tree. In this study we used all of these decision tree algorithms.

Logistic regression is a generalization of linear regression. It is used primarily for predicting binary or multi-class dependent variables. Because the response variable is discrete, it cannot be modeled directly by linear regression. Therefore, rather than predicting a point estimate of the event itself, it builds the model to predict the odds of its occurrence. While logistic regression has been a common statistical tool for classification problems, its restrictive assumptions on normality and independence led to an increased use and popularity of machine learning techniques for real-world prediction problems.

Support Vector Machines (SVMs) belong to a family of generalized linear models which achieves a classification or regression decision based on the value of the linear combination of features. The mapping function in SVMs can be either a classification function (used to categorize the data, as is the case in this study) or a regression function (used to estimate the numerical value of the desired output). For classification, nonlinear kernel functions are often used to transform the input data (inherently representing highly complex nonlinear relationships) to a high dimensional feature space in which the input data becomes more separable (i.e., linearly separable) compared to the original input space. Then, the maximum-margin hyperplanes are constructed to optimally separate the classes in the training data. Two parallel hyperplanes are constructed on each side of the hyperplane that separates the data by maximizing the distance between

Table 2
C5 decision tree from IBM's SPSS data mining tool.

									Bankrupt {Pr}
IF revt ≤76.592	IF Lt ≤13.910	IF dvt≤0	IF rectr ≤5.010	IF prch_f ≤1.400	IF bkvlps ≤0.255	IF gp ≤1.363	IF revt ≤2.888		No {0.913}
"	"	"	"	"	"	"	IF revt>2.888	IF at≤1.516	Yes {1.0}
"	"	"	"	"	"	"	"	IF at>1.516	No {0.833}
"	"	"	"	"	"	IF gp>1.363			No {1.0}
"	"	"	"	"	IF bkvlps>0.255	IF bkvlps≤0.498			Yes {1.0}
"	"	"	"	"	"	IF bkvlps>0.498			No {0.857
"	"	"	"	IF prch_f>1.400					No {0.986}
"	"	"	IF rectr>5.010	IF Lt≤7.767					Yes {0.875}
"	"	"		IF Lt>7.767					No {0.778}
"	"	IF dvt>0							No {1.0}
"	IF Lt>13.910	IF cogs≤0.117							Yes {0.909}
"	"	IF cogs>0.117	IF ebit≤−2.515	IF ni≤3.951	IF rectr≤4.714				No {0.833}
"	"	"	"	"	IF rectr>4.714				Yes {0.692}
"	"	"	"	IF ni>−3.951					Yes {1.0}
"	"	"	IF ebit> −2.515						No {0.888}
IF revt >76.592	IF dvpsx_f ≤0.210	IF Lt ≤251.123	IF invt ≤20.691						Yes {0.967}
"	"	"	IF invt>20.691	IF dvt≤0.047	IF Lt≤197.286	IF prch_f≤6.251			Yes {1.0}
"	"	"	"	"	"	IF prch_f>6.251	IF invt≤54.143	IF cogs≤112.392	No {0.750}
"	"	"	"	"	"	"	"	IF cogs>112.392	No {0.750}
"	"	"	"	"	"	"	IF invt>54.143		No {1.0}
"	"	"	"	"	"	IF prch_f>19.140			Yes {0.923}
"	"	"	"	"	IF Lt>197.286				Yes {1.0}
"	"	"	"	IF dvt>0.047	IF dvt≤1.125				No {0.900}
"	"	"	"	"	IF dvt>1.125				Yes {0.750}
"	"	IF Lt>251.123							Yes {0.978}
"	IF dvpsx_f 0.210	IF rect4r≤237.496	IF at≤1815.05	IF invt≤16.117					Yes {0.909}
"	"	"	"	IF invt>16.117	IF at≤1042.87	IF dvpsx_f≤2.250			No {0.933}
"	"	"	"	"	"	IF dvpsx_f>2.250			Yes {0.833}
"	"	"	"	"	IF at>1042.87				Yes {0.917}
"	"	"	IF at>1815.05						No {1.0}
"	"	IF rectr>237.496							Yes {1.0}

the two parallel hyperplanes. An assumption is made that the larger the margin or distance between these parallel hyperplanes the lower the generalization error of the classifier will be [11].

3.2. Data description

Altman [2] was one of the first researches to predict the corporate bankruptcy. Altman used the classical multivariate discriminate analysis (MDA) technique, which was based on applying the Bayes classification procedure, under the assumption that the two classes have Normal distributions with equal covariance matrices. Altman used the following financial ratios as inputs: working capital/total assets; retained earnings/total assets; earnings before interest and taxes/total assets (ROA); market capitalization/total debt; and sales/total assets (asset turnover). Both the MDA model and the logistic regression model (hereafter LR) have been widely used in practice and in many academic studies. They have been standard benchmarks for the loan default prediction problem. Whereas research studies on

Table 3
CART decision tree from IBM's SPSS data mining tool.

					Bankrupt {Prob}
IF revt≤70.447	IF Lt≤13.914				No {0.917}
	IF Lt>13.914	IF ebit≤0.137	IF ni≤−3.882	IF rectr≤4.798	No {0.821}
				IF rectr>4.798	Yes {0.643}
			IF ni>−3.882		Yes {0.692}
		IF ebit>0.137			No {0.897}
IF revt>70.447	IF Lt≤200.252	IF invt≤29.001			Yes {0.862}
		IF invt>29.001	IF bkvlps≤9.228	IF cogs≤189.341	No {0.696}
				IF cogs>189.341	Yes {0.864}
			IF bkvlps>9.228		No {0.809}
	IF Lt>200.252				Yes {0.927}

Table 4
Comparative correct fits.

Software	Model	Correct classification
IBM SPSS	Logistic regression	0.798
WEKA	"	0.813
IBM SPSS	Neural network (RBF)	0.798
WEKA	"	0.609
IBM SPSS	C5 decision tree	0.937
IBM SPSS	CART decision tree	0.898
WEKA	"	0.932
IBM SPSS	SVM	0.661
WEKA	Best First decision tree	0.929
WEKA	All dimensions decision tree	0.903
WEKA	J48 decision tree	0.948

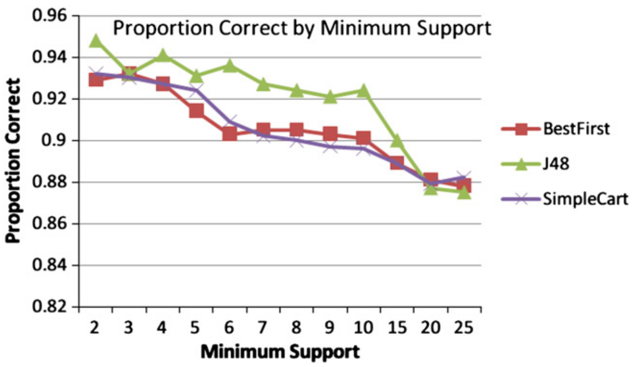

Fig. 3. Relative WEKA decision tree accuracies by minimum support.

using artificial neural network (hence ANN) for bankruptcy prediction started in 1990, and are still active now.

Sample data contained 100 US firms that underwent bankruptcy firms. There were multiple records from different years for the same firm. We first obtained about 400 bankrupt company names using google.com in 2010, and next identified the Ticker name of each company using the Compustat database. We separately obtained data on companies bankrupted during January 2006 and December 2009, since we also wanted to analyze the effects of the economic crisis. After getting the company Ticker code list, obtained financial ratios for the period January 2005 to December 2009 for these firms from the Compustat database. Those financial data and ratios are factors that we can use to predict company bankruptcy. The factors we collected are based on the literature, which contain total asset, book value per share, inventories, liabilities, receivables, cost of goods sold, total dividends, earnings before interest and taxes, gross profit (loss), net income (loss), operating income after depreciation, total revenue, sales, dividends per share, and total market value. In order to compare, we collected the same financial ratios for a roughly equal number of non-failed companies during the same periods. First we used the LexisNexis database to find the company SEC filling after June 2010, which means that companies were still active at the time of this research, and then we selected 100 companies from the results and identified the company CIK code list. Finally we submitted the CIK code list to the Compustat database and got the financial data and ratios during January 2005 to December 2009, which is the same period with that of the failed companies. The reason was to have equal number of samples for bankrupt and non-bankrupt companies.

After all of the data collection and organization, the final data set used in this study consisted of 1321 records with full data over 19 attributes as shown in Table 1. The outcome attribute was bankruptcy, which has a value of 1 if the firm went bankrupt by 2011 (697 cases), and a value of 0 if it did not (624 cases).

Table 5
WEKA decision tree accuracy versus minimum support.

Model	Minimum support	Leaves	Branches	% Correct
Best First	2	46	91	0.899
"	3	39	77	0.893
"	4	34	67	0.891
"	5	26	51	0.889
"	6	20	39	0.886
"	7	18	35	0.877
"	8	16	31	0.874
"	9	14	27	0.874
"	10	12	23	0.870
"	15	4	7	0.869
"	20	8	15	0.869
"	25	8	15	0.871
J48 DT	2	52	103	0.914
"	3	47	93	0.914
"	4	45	89	0.902
"	5	43	85	0.899
"	6	38	75	0.900
"	7	36	71	0.899
"	8	35	69	0.899
"	9	32	63	0.905
"	10	32	63	0.896
"	15	24	47	0.876
"	20	15	29	0.866
"	25	14	27	0.866
CART	2	44	87	0.890
"	3	41	81	0.886
"	4	32	63	0.888
"	5	33	65	0.886
"	6	31	61	0.889
"	7	16	31	0.880
"	8	4	7	0.875
"	9	4	7	0.874
"	10	12	23	0.877
"	15	6	11	0.865
"	20	5	9	0.869
"	25	5	9	0.871

4. Results

The data was modeled using IBM SPSS Modeler (for logistic regression, radial-basis function neural network, C5 and CART decision tree, and support vector machine (SVM) models). WEKA software was used for comparison when models were available, with expanded decision tree tools [45]. WEKA had more decision tree options.

It is obvious that different models had different accuracies, as is to be expected. For this particular set of data, logistic regression was less accurate than decision trees, but more accurate than the radial basis functions run in WEKA. Support vector machines worked less well on this data than any of the other models. Decision tree models were usually better fits for this particular set of data, although neural network model parameters can be set to better fit any particular set of data (at some cost of analyst time).

Results from IBM's SPSS data mining tool indicate that for this specific set of data, the best fit was obtained with the C5 decision tree model (0.937 correct classification). The second best fit was obtained with the CART model (also a decision tree). The support vector machine model had a relatively low correct classification fit of 0.661.

Table 6
WEKA J48 decision tree model results.

Confidence	Minimum support	Leaves	Branches	% Correct	Attributes used
0.10	2	48	95	0.912	15
0.25	"	52	103	0.914	15
0.50	"	52	103	0.914	15
0.10	10	23	45	0.891	12
0.25	"	32	63	0.896	14
0.50	"	32	63	0.896	14
0.10	20	11	21	0.873	6
0.25	"	15	29	0.866	7
0.50	"	15	29	0.868	8

Table 7
CART decision tree from WEKA with minimum support of 5.

											Bankrupt {Pr}
IF revt < 77.696	IF bkvlps < − 3/4292										Yes {1.0}
"	IF bkvlps ≥ − 3.4292	IF cogs < 33.2535	IF ebit < − 12.011	IF bkvlps < 1.5352							No {0.909}
"	"	"	"	IF bkvlps ≥ 1.5352							Yes {0.889}
"	"	"	≥ − 12.011	IF dvt < 0.0005	IF at < 136.1415	IF invt < 5.533	IF prch_f < 1.415	IF cogs < 2.0955			No {0.917}
"	"	"	"	"	"	"	"	IF cogs ≥ 2.0955	IF cogs < 3.084	IF invt < 0.491	Yes {1.0}
"	"	"	"	"	"	"	"	"	"	IF invt ≥ 0.491	No {0.833}
"	"	"	"	"	"	"	"	"	IF cogs ≥ 3.084	IF bkvlps < −.25935	No {0.925}
"	"	"	"	"	"	"	"	"	"	IF bkvlps ≥ −.25935	Yes {0.571}
"	"	"	"	"	"	"	IF prch_f ≥ 1.415				No {0.974}
"	"	"	"	"	"	IF invt ≥ 5.533	IF Lt < 8.085				Yes {0.875}
"	"	"	"	"	"	"	IF Lt ≥ 8.085				No {1.0}
"	"	"	"	"	IF at ≥ 136.1415						Yes {0.75}
"	"	"	"	IF dvty ≥ 0.0005							No {0.988}
"	"	IF cogs ≥ 33.2535	IF ebit < 1.041	IF ni < − 6.717	IF rnd < 0.37109						Yes {0.800}
"	"	"	"	"	IF rnd ≥ 0.37109						No {0.900}
"	"	"	"	IF ni ≥ − 6.717							Yes {0.909
"	"	"	IF ebit ≥ 1.041								No {1.0}
IF revt ≥ 77.696	IF dvpsx_f < 0.215	IF Lt < 251.53	IF invt < 60.0275	IF revt < 120.2765	IF ebit < 1.1055						Yes {1.0}
"	"	"	"	"	IF ebit ≥ 1.1055						No {0.714}
"	"	"	"	IF revt ≥ 120.2765							Yes {0.957}
"	"	"	IF invt ≥ 60.0275	IF rectr < 10.746							Yes {1.0}
"	"	"	"	IF rectr ≥ 10.746	IF prch_f < 18.9	IF Lt < 206.625					No {1.0}
"	"	"	"	"	"	IF Lt ≥ 206.625					Yes {0.600}
"	"	"	"	"	IF prch_f ≥ 18.9						Yes {0.857}
"	"	IF Lt ≥ 251.53	IF revt < 293.7785	IF fyear < 2006.5							Yes {1.0}
"	"	"	"	IF fyear ≥ 2006.5							No {0.700}
"	"	"	IF revt ≥ 293.7785								Yes {0.995}
"	IF dvpsx_f ≥ 0.215	IF dvt < 32.381	IF invt ≥ 13.513	IF at < 827.2125							Yes {1.0}
"	"	"	"	IF at ≥ 827.2125							No {0.833}
"	"	"	IF invt ≥ 13.513	IF at < 1043.2265							No {0.913}
"	"	"	"	IF at ≥ 1043.2265	IF Lt < 1320.134						Yes {1.0}
"	"	"	"	"	IF Lt ≥ 1320.134						No {1.0}
"	"	IF dvt ≥ 32.381									Yes {0.948}

Table 8
CART decision tree from WEKA with minimum support of 9.

			Yes	No	Bankrupt {Prob}
IF revt < 77.696			80	522	No {0.867}
IF revt ≥ 77.696	IF dvpsx_f < 0.215		531	42	Yes {0.927}
	IF dvpsx_f ≥ 0.215	IF dvt < 32.381	31	57	No {0.648}
"	"	IF dvt ≥ 32.381	55	3	Yes {0.948}

Replicating the analysis with WEKA tools attain found the best fits for decision tree models (the standard J48 decision tree model yielding the overall highest accuracy). Of course, model fit varies widely across data sets, and fine tuning either decision trees or neural network models can improve classification accuracy. But the primary advantage of decision trees is their transparency.

The IBM SPSS Modeler's C5 model involved a rather complicated set of rules, using 13 of the 18 available variables in 30 branches with 31 decision-concluding nodes. The decision tree is shown in Table 2.

The length of decision trees can be controlled by varying minimum support and pruning, so we are not claiming generalizable differences in the number of rules by method. But we are trying to demonstrate the complexity involved for human application in having too many rules.

Table 3 shows the much shorter model obtained from CART for this data.

The proportion of "No" results for the population was 624 out of 1321 cases, or a proportion of 0.472. The model had an average correct classification rate (after 10-fold cross validation sampling) of 0.898. Five variables were used in ten rows of rules. The CART model had a slightly inferior fit (0.898 correct classification versus 0.937 for the C5 model), but required only 9 branches with ten decision-concluding nodes based upon 8 variables. While this is less than the 0.937 obtained with the C5 model, it is more concise, and therefore more usable. The C5 model involved 13 variables with 31 rows of rules. The more concise model requires users to find less data, and is expected to involve less uncertainty. Table 4 compares model fits for IBM SPSS and WEKA models, all using 1321 observations:

A WEKA J48 model used 46 rows of rules involving 13 variables. Thus the J48 model was even less concise than the C5 model from IBM's SPSS tool. The J48 model did correctly classify 0.948 of the cases tested. However, this again demonstrates the tradeoff between model accuracy and transparency/transportability.

Our interest is in comparing the decision tree models. WEKA provides the ability to adjust the parameter for minimum support. Larger minimum support requirements yield fewer rules. Table 5 and Fig. 3

Table 9
J48 decision tree from WEKA with minimum support of 15.

								Bankrupt {Pr} total/error
IF revt ≤ 76.592	IF Lt ≤ 13.91	IF dvt ≤ 0	IF rectr ≤ 5.01	IF fyear ≤ 2007	IF prch_f ≤ 1.4	IF cogs ≤ 2.089		No {0.894} 84/10
"	"	"	"	"	"	IF cogs > 2.089	IF gp ≤ 1.76	Yes {0.800} 20/5
"	"	"	"	"	"	"	IF gp > 1/76	No {0.889} 24/3
"	"	"	"	"	IF prch_f > 1.4			No {0.982} 110/2
"	"	"	"	IF fyear > 2.007				No {1.0} 77/0
"	"	"	IF rectr > 5.01					Yes {0.680} 17/8
"	"	IF dvt > 0						No {1.0} 125/0
"	IF Lt > 13.91	IF revt ≤ 8.73						Yes {0.733} 22/6
"	"	IF revt > 8.73	IF dvpsx_f ≤ 0.01	IF at ≤ 139.611				No {0.871} 61/9
"	"	"	"	IF at > 139.611				Yes {0.714} 25/10
"	"	"	IF dvpsx_f > 0.01					No {0.974} 37/1
IF revt > 76/592	IF dvpsx_f ≤ 0.21	IF Lt ≤ 251.123	IF invt ≤ 20.691					Yes {0.968} 61/2
"	"	"	IF invt > 20.691	IF dvt ≤ 0.047	IF Lt ≤ 197.286	IF prch_f ≤ 6.251		Yes {1.0} 15/0
"	"	"	"	"	"	IF prch_f > 6.251	IF mkvalt ≤ 86.9214	No {0.800} 16/4
"	"	"	"	"	"	"	IF mkvalt > 86/9214	Yes {0/771} 27/8
"	"	"	"	"	IF Lt > 197.286			Yes {1.0} 22/0
"	"	"	"	IF dvt > 0.047				No {0.720} 18/7
"	"	IF Lt > 251.123						Yes {0.979} 414/9
"	IF dvpsx_f > 0.21	IF rectr ≤ 237.496	IF Lt ≤ 1319.934	IF invt ≤ 16.117				Yes {0.917} 22/2
"	"	"	"	IF invt > 16.117	IF invt ≤ 72.857			No {1.0} 24/0
"	"	"	"	"	IF invt > 72.857	IF mkvalt ≤ 465.9829		Yes {0.889} 16/2
"	"	"	"	"	"	IF mkvalt > 465.9829		No {0.815} 22/5
"	"	"	IF Lt > 1319.934					No {1.0} 15/0
"	"	IF rectr > 237.496						Yes {1.0} 47/0

Table 10
J48 decision tree from WEKA with minimum support of 20.

						Bankrupt {Pr} total/error
IF revt ≤ 76.592	IF Lt ≤ 13.91					No {0.921} 457/39
"	IF Lt > 13.91	IF revt ≤ 0.873				Yes {0.786} 22/6
"	"	IF revt > 8.73	IF dvpsx_f ≤ 0.01	IF at ≤ 139.611		No {0.871} 61/9
"	"	"	"	IF at > 139.611		Yes {0.714} 25/10
"	"	"	IF dvpsx_f > 0.01			No {0.974} 37/1
IF revt > 76.592	IF dvpsx_f ≤ 0.21	IF Lt ≤ 251.123	IF invt ≤ 20.691			Yes {0.968} 61/2
"	"	"	IF invt > 20.691	IF ni ≤ −3.252		Yes {0.889} 32/4
"	"	"	"	IF ni > −3.252	IF Lt ≤ 88.296	No {0.735} 25/9
"	"	"	"	"	IF Lt > 88.296	Yes {0.788} 41/11
"	"	IF Lt > 251.123				Yes {0.979} 414/9
"	IF dvpsx_f > 0.21	IF rectr ≤ 237.496	IF gp ≤ 254.644	IF dvpsx_f ≤ 0.69	IF gp ≤ 86.628	No {0.906} 29/3
"	"	"	"	"	IF gp > 86.628	Yes {0.686} 24/11
"	"	"	"	IF dvpsx_f > 0.69		Yes {0.889} 24/3
"	"	"	IF gp > 254.644			No {0.917} 22/2
"	"	IF rectr > 237.496				Yes {1.0} 47/0

Table 11
J48 decision tree from WEKA with minimum support of 25.

						Bankrupt {Pr} total/error
IF revt ≤ 76.592	IF Lt ≤ 13.91					No {0.940} 457/39
"	IF Lt > 13.91	IF ebit ≤ 1.015	IF invt ≤ 1.421	IF ni ≤ −4.884		No {0.694} 25/11
"	"	"	"	IF ni > −4.884		Yes {0. 862} 25/4
"	"	"	IF invt > 1.421			No {0.868} 33/5
"	"	IF ebit > 1.015				No {0.939} 62/4
IF revt > 76.592	IF dvpsx_f ≤ 0.21	IF Lt ≤ 251.123	IF invt ≤ 20.691			Yes {0.968} 61/2
"		"	IF invt > 20.691	IF ni ≤ −3.252		Yes {0.889} 32/4
"	"	"	"	IF ni > −3.252	IF Lt ≤ 88.296	No {0.735} 25/9
"	"	"	"	"	IF Lt > 88.296	Yes {0.788} 41/11
"	"	IF Lt > 251.123				Yes {0.979} 414/9
"	IF dvpsx_f > 0.21	IF rectr ≤ 237.496	IF gp ≤ 239.833	IF revt ≤ 353.155		No {0.865} 32/5
"	"	"	"	IF revt > 353.155		Yes {0.792} 42/11
"	"	"	IF gp > 239.833			No {0.893} 25/3
"	"	IF rectr > 237.496				Yes {1.0} 47/0

Table 12
BFTree MS = 10.

					Bankrupt {Pr} total/error
IF revt < 77.696					No {0.867} 522/80
IF revt ≥ 77.696	IF dvpsx_f < 0.215	IF Lt < 251.53	IF invt < 60.0275		Yes {0.878} 101/14
"	"	"	IF invt ≥ 60.0275	IF rectr < 10.746	Yes {1.0} 13/0
"	"	"	"	IF rectr ≥ 10.746	No {0.613} 19/12
"	"	IF Lt ≥ 251.53	IF revt < 293.7785	IF fyear < 2006.5	Yes {1.0} 30/0
"	"	"	"	IF fyear ≥ 2006.5	No {0.700} 7/3
"	"	"	IF revt ≥ 293.7785		Yes {0.995} 372/2
"	IF dvpsx_f ≥ 0.215	IF dvt < 32.381	IF invt < 13.513	IF at < 827.2125	Yes {1.0} 14/0
"	"	"	"	IF at ≥ 827.2125	No {0.833} 10/2
"	"	"	IF invt ≥ 13.513	IF at < 1043.2265	No {0.913} 42/4
"	"	"	"	IF at ≥ 1043.2265	Yes {0.687} 11/5
"	"	IF dvt ≥ 32.381			Yes {0.948} 55/3

Table 13
BFTree MS = 15.

					Bankrupt {Pr} total/error
IF revt < 77.696					No {0.867} 522/80
IF revt ≥ 77.696	IF dvpsx_f < 0.215				Yes {0.927} 531/42
"	IF dvpsx_f ≥ 0.215	IF dvt < 32.381	IF invt < 13.513		Yes {0.615} 16/10
"	"	"	IF invt ≥ 13.513	IF at < 1043.2265	No {0.913} 42/4
"				IF at ≥ 1043.2265	Yes {0.687} 11/5
"	"	IF dvt ≥ 32.381			Yes {0.948} 55/3

Table 14
BFTree MS = 20.

				Bankrupt {Pr} total/error
IF revt < 77.696				No {0.867} 522/80
IF revt ≥ 77.696	IF dvpsx_f < 0.215			Yes {0.927} 531/42
"	IF dvpsx_f ≥ 0.215	IF dvt < 32.381	IF invt < 13.513	Yes {0.615} 16/10
"	"	"	IF invt ≥ 13.513	No {0.758} 47/15
"	"	IF dvt ≥ 32.381		Yes {0.948} 55/3

show the number of rules (decision tree leaves) and accuracy for different WEKA decision tree models.

The WEKA J48 model seemed to provide superior fit (at least for smaller minimum support levels). This model has a confidence parameter that can be set in addition to minimum support. We ran models as shown in Table 6 to see the impact of confidence factor. We chose minimum support levels of 2, 10, and 20 to get diverse settings after seeing the results shown in Table 5.

Table 15
BFTree MS = 25.

			Bankrupt {Pr} total/error
IF revt < 77.696			No {0.847} 522/80
IF revt ≥ 77.696	IF dvpsx_f < 0.215		Yes {0.921} 531/42
"	IF dvpsx_f ≥ 0.215	IF dvt < 32.381	No {0.456} 57/31
"	"	IF dvt ≥ 32.381	Yes {0.945} 55/3

The confidence factor does not seem to have a big impact. For minimum support of 2, confidence settings of 0.25 and 0.50 yielded precisely the same model (although the proportion correct reported varied slightly). Using minimum support of 10 yielded the same model with all three confidence levels. For minimum support of 20, confidence settings of 0.25 and 0.50 models were identical. Thus we used the default confidence setting of 0.25.

4.1. Tradeoff — fidelity and transparency

It can be seen that the number of decision rules (leaves) tends to decrease with higher minimum support, while correct classification also decreases with higher minimum support. The more rules, the more difficult to apply (transport) the model. Thus there is a tradeoff between accuracy and transportability. For instance, the CART model obtained from WEKA with minimum support of 9 yielded the set of rules given in Table 7. The last column shows the rule outcome, and the proportion correct over the training set.

This model involved 22 concluding leaves with 11 attributes. It had an accuracy of 0.924 on average over the 10-fold testing. The CART model with minimum support set at 9 is given in Table 8:

There is a clear advantage for the last set of decision rules obtained with minimum support of 9, with only 12 leaves involving 7 attributes, versus the case where minimum support was 5 (22 leaves involving 11 attributes. The tradeoff in lost accuracy is slight — the model with minimum support of 5 tested at 0.924 accuracy, versus 0.897 for the model with minimum support of 9. Tables 8, 9 and 10 show the resulting decision rules, with greater transportability attained from higher minimum support levels. Tables 9, 10 and 11 give decision tree rules for WEKA J48 decision trees with varying minimum support levels. The last column shows the rule outcome, and the proportion correct over the training set. The total number of cases for each rule is given separated by "/" from the erroneous cases.

The correct classifications for these three decision trees are fairly similar (0.900 for MS = 15, 0.877 for MS = 20, 0.875 for MS = 25). The number of rules dropped from 21 for the MS = 15 model to 16 for the MS = 20 model to 10 for the MS = 25 model. The number of attributes ranged from 8 for MS = 25, 7 for MS = 20, and 11 for MS = 15. Clearly the MS = 25 model is easier to implement.

The Best First Decision Tree models showed a similar pattern, as shown in Tables 12, 13, 14, and 15:

Again, there is a clear reduction in rule size with higher minimum support. The number of variables included in the decision tree will tend to decline with higher minimum support as well (see Appendix A). There is usually a cost in terms of correct classification, but fewer attributes leave less error potential from having to estimate attribute values in application.

5. Conclusions

Any particular set of data will have different relative fits from different data mining models. That is why it is conventional to apply logistic regression, neural networks, and decision trees to data. Neural network models often provide very good fit with a particular data set, but they are not transparent nor easily transportable. Decision tree models are expressed in easily understood terms. A common problem with decision trees is that models generate too many rules. This can be controlled by increasing the minimum support required for a rule.

Our study demonstrated this point. For this data, the overall best fit (0.948 average accuracy) was obtained with a WEKA J48 decision tree model with minimum support of 2. However, that involved 46 leaves to the decision tree. Even setting the minimum support to 9 yielded 28 leaves to the tree involving 11 attributes (average accuracy dropping to 0.921). While the set of rules is transportable and transparent, it is bulky and complex. A decision tree obtained from a WEKA CART model with minimum support of 9 yielded a model we would argue was preferable, involving 12 leaves to the tree and 7 attributes. There was degradation in average testing accuracy (dropping to 0.897).

The particular choice would depend upon user preferences. Our point is that there is a tradeoff between average accuracy and decision tree size that can be controlled through the minimum support parameter.

Appendix A. J48 model results

	MS2	MS3	MS4	MS5	MS6	MS7	MS8	MS9	MS10	MS15	MS20	MS25
fyear	x	x	x	x	x	x	x	x	x			
cik	x	x	x	x	x	x	x	x	x	x	x	x
at	x	x	x	x	x	x	x			x	x	
bkvlps	x	x	x	x	x	x	x	x	x	x		
invt	x	x	x	x	x	x	x	x	x	x		x
Lt	x	x	x	x	x	x	x	x	x	x	x	x
rectr	x	x	x	x	x	x	x	x	x	x	x	x
cogs		x	x	x								
dvt	x	x	x	x	x	x				x	x	
ebit	x	x	x	x	x	x		x	x	x		x
gp				x						x		
ni												x
oiadp												
revt	x	x	x	x	x	x	x	x	x	x	x	x
sale												
dvpsx_f	x	x	x	x	x	x	x	x	x	x	x	x
mkvalt	x	x	x	x	x	x	x	x	x			
prch_f												
Bankruptcy												
# attributes	12	13	13	14	12	12	10	10	10	11	7	8
Leaves	46	41	38	34	31	31	29	28	28	21	16	10
Tree size	91	81	75	67	61	61	57	55	55	41	31	19
Correct	0.948	0.943	0.942	0.931	0.936	0.927	0.924	0.921	0.923	0.9	0.877	0.875

References

[1] E. Alfaro, N. García, M. Gámez, D. Elizondo, Bankruptcy forecasting: an empirical comparison of AdaBoost and neural networks, Decision Support Systems 45 (1) (2008) 110–122.
[2] E.I. Altman, Financial ratios, discriminant analysis and the prediction of corporate bankruptcy, Journal of Finance 23 (4) (1968) 589–609.
[3] A.F. Atiya, Bankruptcy prediction for credit risk using neural networks: a survey and new results, IEEE Transactions on Neural Networks 12 (4) (2001) 929–935.
[4] N. Barakat, A.P. Bradley, Rule extraction from support vector machines: a review, Neurocomputing 74 (1–3) (2010) 178–190.
[5] W.H. Beaver, Financial ratios as predictors of failure, Journal of Accounting Research 4 (3) (1966) 71–111.
[6] D. Berg, bankruptcy prediction by generalized additive models, Applied Stochastic Models in Business and Industry 23 (2) (2007) 129–143.
[7] L. Breiman, J.H. Friedman, R.A. Olshenm, C.J. Stone, Classification and Regression Trees, Wadsworth & Brooks/Cole Advanced Books & Software, Monterey, CA, 1984.
[8] C.W. Chan, W.C. Chan, A.W. Jayawardena, C.J. Harris, Structure selection of neurofuzzy networks based on support vector regression, International Journal of Systems Science 33 (9) (2002) 715–722.
[9] X. Chen, X. Wang, D.D. Wu, Credit risk measurement and early warning of SMEs: an empirical study of listed SMEs in China, Decision Support Systems 49 (3) (2010) 301–310.
[10] S. Cho, H. Hong, B.-C. Ha, A hybrid approach based on the combination of variable selection using decision trees and case-based reasoning using the Mahalanobis distance: for bankruptcy prediction, Expert Systems with Applications 37 (4) (2010) 3482–3488.
[11] N. Cristianini, J. Shawe-Taylor, An Introduction to Support Vector Machines and other Kernel-based Learning Methods, Cambridge University Press, London, 2000.
[12] D. Delen, A comparative analysis of machine learning techniques for student retention management, Decision Support Systems 49 (4) (2010) 498–506.
[13] S. Haykin, Neural Networks and Learning Machines, 3 rd Ed. Prentice Hall, New Jersey, 2008.
[14] K. Hornik, M. Stinchcombe, H. White, Universal approximation of an unknown mapping and its derivatives using multilayer feedforward network, Neural Networks 3 (1990) 359–366.
[15] Z. Huang, H. Chen, C.-J. Hsu, W.-H. Chen, S. Wu, Credit rating analysis with support vector machines and neural networks: a market comparative study, Decision Support Systems 37 (4) (2004) 543–558.
[16] R. Kohavi, A study of cross-validation and bootstrap for accuracy estimation and model selection, in: S. Wermter, E. Riloff, G. Scheler (Eds.), The Fourteenth International Joint Conference on Artificial Intelligence (IJCAI) Montreal, Quebec, Canada, Morgan Kaufman, San Francisco, CA, 1995, pp. 1137–1145.
[17] D. Kukolj, E. Levi, Identification of complex systems based on neural and Takagi–Sugeno fuzzy model, IEEE Transactions on Systems, Man & Cybernetics: Part B 34 (1) (2004) 272–282.
[18] M. Lam, Neural network techniques for financial performance prediction: integrating fundamental and technical analysis, Decision Support Systems 37 (4) (2004) 567–581.
[19] W. Leigh, R. Purvis, J.M. Ragusa, Forecasting the NYSE composite index with technical analysis, pattern recognizer, neural network, and genetic algorithm: a case study in romantic decision support, Decision Support Systems 32 (4) (2002) 361–377.
[20] K. Li, J.-X. Peng, System oriented neural networks — problem formulation, methodology and application, International Journal of Pattern Recognition & Artificial Intelligence 20 (2) (2006) 143–158.
[21] D. Martens, L. Bruynseels, B. Baesens, M. Willekens, J. Vanthienen, Predicting going concern opinion with data mining, Decision Support Systems 45 (4) (2008) 765–777.
[22] C.R. Milaré, A.C.P. de Carvalho, M.C. Monard, An approach to explain neural networks using symbolic algorithms, International Journal of Computational Intelligence and Applications 2 (4) (2002) 365–376.
[23] A. Nachev, S. Hill, C. Barry, B. Stoyanov, Fuzzy, distributed, instance counting, and default Artmap neural networks for financial diagnosis, International Journal of Information Technology and Decision Making 9 (6) (2010) 959–978.
[24] J. Neves, A. Vieira, Improving bankruptcy prediction with hidden layer learning vector quantization, The European Accounting Review 15 (2) (2006) 253–271.
[25] S. Önsei, F. Ülengin, G. Ulusoy, E. Aktaş, Ö. Kabak, Y.I. Topcu, A new perspective on the competitiveness of nations, Socio-Economic Planning Sciences 42 (4) (2008) 221–246.
[26] W. Pedrycz, Heterogeneous fuzzy logic networks: fundamentals and development studies, IEEE Transactions on Neural Networks 15 (6) (2004) 1466–1481.
[27] P. Pendharkar, Misclassification cost minimizing fitness functions for genetic algorithm-based artificial neural network classifiers, Journal of the Operational Research Society 60 (8) (2009) 1123–1134.
[28] L. Quinlan, Induction of decision trees, Machine Learning 1 (1986) 81–106.
[29] L. Quinlan, C4.5: Programs for Machine Learning, Morgan Kaufmann, San Mateo, CA, 1993.
[30] P. Ravi Kumar, V. Ravi, Bankruptcy prediction in banks and firms via statistical and intelligent techniques — a review, European Journal of Operational Research 180 (1) (2007) 1–18.
[31] R. Risser, C. Chaloupka, W. Grundler, M. Sommer, J. Häusler, C. Kaufmann, Using non-linear methods to investigate the criterion validity of traffic-psychological test batteries, Accident Analysis and Prevention 40 (1) (2008) 149–157.
[32] Y.U. Ryu, W.T. Yue, Firm bankruptcy prediction: experimental comparison of isotonic separation and other classification approaches, IEEE Trnasactions on Systems, Man & Cybernetics: Part A 35 (5) (2005) 727–737.
[33] T. Sen, P. Ghandforoush, C.T. Stivason, Improving prediction of neural networks: a study of two financial prediction tasks, Journal of Applied Mathematics and Decision Sciences 8 (4) (2004) 219–233.
[34] J.R. Shah, M.B. Murtaza, A neural network based clustering procedure for bankruptcy prediction, American Business Review 18 (2) (2000) 80–86.
[35] K.-S. Shin, T.S. Lee, H.-J. Kim, An application of support vector machines in bankruptcy prediction model, Expert Systems with Applications 28 (1) (2005) 127–135.
[36] A.P. Sinha, H. Zhao, Incorporating domain knowledge into data mining classifiers: an application in indirect lending, Decision Support Systems 46 (1) (2008) 287–299.
[37] D. Stathakis, A. Vasilakos, Comparison of computational intelligence based classification techniques for remotely sensed optimal image classification, IEEE Transactions on Geoscience and Remote Sensing 44 (8) (2006) 2305–2318.
[38] T. Sueyoshi, G.R. Tadiparthi, An agent-based decision support system for wholesale electricity market, Decision Support Systems 44 (2) (2008) 425–446.
[39] T.K. Sung, N. Chang, G. Lee, Dynamics of modeling in data mining: interpretive approach to bankruptcy prediction, Journal of Management Information Systems 16 (1) (1999) 63–85.
[40] C.-F. Tsai, Financial decision support using neural networks and support vector machines, Expert Systems 25 (4) (2008) 380–393.
[41] C.-F. Tsai, Y.-C. Hsiao, Combining multiple feature selection methods for stock prediction: union, intersection, and multi-intersection approaches, Decision Support Systems 50 (1) (2010) 258–269.
[42] F.-M. Tseng, Y.-C. Hu, Comparing four bankruptcy prediction models: logit, quadratic interval logit, neural and fuzzy neural networks, Expert Systems with Applications 37 (3) (2010) 1846–1853.
[43] D. West, S. Dellana, J. Qian, Neural network ensemble strategies for financial decision applications, Computers and Operations Research 32 (10) (2005) 2543–2559.
[44] R.L. Wilson, R. Sharda, Bankruptcy prediction using neural networks, Decision Support Systems 11 (5) (1994) 545–557.
[45] I.H. Witten, E. Frank, Data Mining: Practical Machine Learning Tools and Techniques, Morgan Kauffman, San Francisco, 2005.
[46] Z.R. Yang, M.B. Platt, H.D. Platt, Probabilistic neural networks in bankruptcy prediction, Journal of Business Research 44 (2) (1999) 67–74.
[47] X. Yuan, Y. Wang, L. Wu, X. Zhang, W. Sun, Neural network based self-learning control strategy for electronic throttle valve, IEEE Transactions on Vehicular Technology 59 (8) (2010) 3757–3765.
[48] G. Zhang, M.Y. Hu, B.E. Patuwo, D.C. Indro, Artificial neural networks in bankruptcy prediction: general framework and cross-validation analysis, European Journal of Operational Research 116 (1) (1999) 16–32.
[49] H. Zhao, A multi-objective genetic programming approach to developing Pareto optimal decision trees, Decision Support Systems 43 (3) (2007) 809–826.
[50] Z.-H. Zhou, Y. Jiang, S.-F. Chen, Extracting symbolic rules from trained neural network ensembles, AI Communications 16 (2003) 3–15.

Dr. David L. Olson is the James & H.K. Stuart Professor in MIS and Chancellor's Professor at the University of Nebraska. He has published research in over 100 refereed journal articles, primarily on the topic of multiple objective decision-making and information technology. He teaches in the management information systems, management science, and operations management areas. He has authored 17 books, to include *Decision Aids for Selection Problems*, *Introduction to Information Systems Project Management*, and *Managerial Issues of Enterprise Resource Planning Systems* as well as co-authored the books *Introduction to Business Data Mining*, *Enterprise Risk Management*, *Advanced Data Mining Techniques*, *New Frontiers in Enterprise Risk Management*, *Enterprise Information Systems*, and *Enterprise Risk Management Models*. He is associate editor of *Service Business* and co-editor in chief of *International Journal of Services Sciences*. He has made over 100 presentations at international and national conferences on research topics. He is a member of the Decision Sciences Institute, the Institute for Operations Research and Management Sciences, and the Multiple Criteria Decision Making Society. He was a Lowry Mays endowed Professor at Texas A&M University from 1999 to 2001. He was named the Raymond E. Miles Distinguished Scholar award for 2002, and was a James C. and Rhonda Seacrest Fellow from 2005 to 2006. He was named Best Enterprise Information Systems Educator by IFIP in 2006. He is a Fellow of the Decision Sciences Institute.

Dr. Dursun Delen is an Associate Professor of Management Science and Information Systems in the Spears School of Business at Oklahoma State University (OSU). He received his Ph.D. in Industrial Engineering and Management from OSU in 1997. Prior to his appointment as an Assistant Professor at OSU in 2001, he worked for a private consultancy company, Knowledge Based Systems Inc., in College Station, Texas, as a research scientist for five years, during which he led a number of decision support and other information systems related research projects funded by federal agencies such as DoD, NASA, NIST and DOE. His research has appeared in major journals including Decision Support Systems, Communications of the ACM, Computers and Operations Research, Computers in Industry, Journal of Production Operations Management, Artificial Intelligence in Medicine, Expert Systems with Applications, among others. He has recently co-authored three books on data mining, decision support systems and business intelligence. He served as the general co-chair for the 4th International Conference on Network Computing and Advanced Information Management, and is regularly organizing tracks and mini-tracks for several international conferences. Dr. Delen serves on several technical journal editorial boards as associate editor-in-chief, associate editor and editorial board member. His research interests are in decision support systems, data/text mining, knowledge management, business intelligence and enterprise modeling.

Yanyan Meng is a Master's student in management information systems in the College of Business Administration, University of Nebraska – Lincoln.