Journal of International Technology and Information Management

Volume 23 | Issue 1

Article 5

2014

The Classification Performance of Multiple Methods and Datasets: Cases from the Loan Credit Scoring Domain

Jozef Zurada
University of Louisville

Niki Kunene
University of Louisville

Jian Guan
University of Louisville

Recommended Citation

Zurada, Jozef; Kunene, Niki; and Guan, Jian (2014) "The Classification Performance of Multiple Methods and Datasets: Cases from the Loan Credit Scoring Domain," *Journal of International Technology and Information Management*: Vol. 23: Iss. 1, Article 5.
Available at: http://scholarworks.lib.csusb.edu/jitim/vol23/iss1/5

The Classification Performance of Multiple Methods and Datasets: Cases from the Loan Credit Scoring Domain

Jozef Zurada
Niki Kunene
Jian Guan
Department of Computer Information Systems
University of Louisville
USA

ABSTRACT

Decisions to extend credit to potential customers are complex, risky and even potentially catastrophic for the credit granting institution and the broader economy as underscored by credit failures in the late 2000s. Thus, the ability to accurately assess the likelihood of default is an important issue. In this paper the authors contrast the classification accuracy of multiple computational intelligence methods using five datasets obtained from five different decision contexts in the real world. The methods considered are: logistic regression (LR), neural network (NN), radial basis function neural network (RBFNN), support vector machine (SVM), k-nearest neighbor (kNN), and decision tree (DT). The datasets have various characteristics with respect to the number of cases, the number and type of attributes, the extent of missing values as well as different ratios for bad loans/good loans. Using areas under ROC charts as well as the classification accuracy rates for overall, bad loans, and good loans the performances of six methods across five datasets and the five datasets across the methods are examined to find if there are significant differences between the methods and datasets. Our results reveal some interesting findings which may be useful to practitioners. Even though no method consistently outperformed any other method using the above metrics on all datasets, this study provides some guidelines as to the most appropriate methods suitable for each specific data set. In addition, the study finds that customer financial attributes are much more relevant than the personal, social, or employment attributes for predictive accuracy.

INTRODUCTION

The great recession of the late 2000s has re-focused people's attention on the risk of credit extension as an engine of global economic activity. The bust of the housing market and the defaults of subprime mortgages extended to borrowers with weak credit precipitated an implosion of the mortgage backed securities and collateralized debt obligations industry (Lim, 2008). The consequences resulting from creditors' failure, as well as the failure of regulators to accurately assess the credit risk of potential borrowers, had a catastrophic impact on the global financial system and broader economic activity. Credit scoring models are tools used to assess the likelihood of a potential debtor defaulting on a credit arrangement, allowing the creditor to determine whether to enter into a credit arrangement. These models have also been used by regulators to retrospectively assess credit agreements with profound impacts on an industry or economy. In general, credit-scoring models require that debtors be classified into groups of *good credit* (least risky) and *bad credit* (most risky). An ability to correctly classify a debtor has broad financial implications for credit granting institutions and the economy. Studies show that even a 1% improvement in the accuracy of a credit scoring model can save millions of dollars in a large loan portfolio (West, 2000). For modern economies where credit availability is central to economic activity, reliable credit-scoring models are an imperative.

Credit scoring models have a history spanning decades within lending institutions (West, 2000). The research on credit scoring models has used a variety of analytical methods, including statistical and data mining methods and on a variety of datasets. These methods include: survival analysis (Stepanova & Thomas, 2002) which is used to predict the time to default, or time to early repayment, linear discriminant analysis (Bikakis et al., 2009), logistic regression (LR) (Sohn & Kim, 2007), *k*-nearest neighbor (*k*NN) (Laha, 2007), classification trees (Urdaneta et al.), neural networks (NN) (Khashman, 2009; West, 2000), radial basis function neural networks (RBFNN), support vector machines (SVM) (Belloti & Crook, 2009; Chen, Ma, & Ma, 2009; Li, Shiue, & Huang, 2006; Luo, Cheng, & Hsieh, 2009; Tsai, 2008; Zhou, Lai, & Yu, 2008), decision trees (DT) (Owen, 2003; Zurada, 2007, 2010), ensemble techniques (Chrzanowska et al., 2009; West et al., 2005), and genetic programming (Espejo, Ventura, & Herrera, 2010; Finlay, 2009; Huang, Tzeng, & Ong, 2006; Ong, Huang, & Tzeng, 2005). In these and related studies, models are typically benchmarked, and the comparison of multiple models with respect to accuracy (Baesens, Setiono, Mues, & Vanthienen, 2003) is a regular feature. However, such studies frequently employ a single dataset. Comparisons based on a single dataset are limited by the inevitable idiosyncrasies of the dataset, its context, as well as the chosen computational method. Therefore studies that examine the performances of different methods on different datasets are important to help us better understand the relative strengths of different methods and the characteristics of datasets. To the best of our knowledge no credit scoring study has undertaken an in-depth comparative examination of these computational methods within the context of different data settings. This paper describes a carefully designed study to assess the effectiveness of several different methods on a collection of datasets from different contexts. In this study we use five datasets obtained from varying contexts to compare six methods. The datasets are Australian, SAS-1, SAS-2, German, and Farmer datasets. The six methods are: logistic regression (LR), neural network (NN), radial basis function neural network (RBFNN), support vector machine (SVM), *k*-nearest neighbor (*k*NN), and decision tree (DT).

In this study, when methods are applied to data and its context, they are defined as models. This distinction is consistent with the design science tradition of March and Smith (1995) and (Hevner, March, Park, & Ram, 2004). March and Smith (1995) describe methods as algorithms and practices. Methods "define processes…they define how to …search the solution space; on the other hand, models represent a real world situation, i.e., the design problem and the solution space" (March & Smith, 1995). Our results, therefore, refer to models rather than methods.

In a guide to IS researchers on what constitutes a contribution, Zmud (2013) includes the following as a contribution: providing "new insights into why, when, and where of a phenomenon (i.e. drilling down inside the black box)." In this study, our results offer richer interpretation because not only is each model assessed against multiple datasets, but model performance on each dataset is also assessed using multiple modes. Model performance is evaluated not only using the common probability 0.5 cutoff point, but also using the area under receiver operating characteristic (ROC) curves and the curves themselves to determine the overall efficiency of the models, or look at more specific classification accuracy levels at various operating cut-off points. As a result, the findings in this study are more nuanced, frequently reflecting that a model's performance cannot be said to be simply universally better or worse than others. There are wheres and whens.

For example, our results show, with respect to dataset quality: The largeness of a dataset is not an unqualified positive performance characteristic. A dataset with only continuous variables performs poorly, even though real numeric variables are most important to the classification problem. A multidimensional dataset, i.e. with more attributes, doesn't necessarily mean better performance, but more balanced datasets perform better overall. With respect to model performance: SVM performs best on the more balanced datasets using global performance metrics, whereas NN and DTs do very well on an unbalanced dataset with missing values. *k*NN does best on an unbalanced dataset without missing values using global performance metrics. DTs perform relatively better and certainly no worse than other models on average bad loan classification at the 0.5 cutoff performing especially better on the unbalanced datasets. The latter is a different, but more useful, finding than a prior study's finding that NNs perform best on bad loans (Chen and Huang (2003), where the only dataset used was a balanced dataset. Finally, we also find that, for practitioners making data collection decisions in this area, customer financial attributes like the debt-ratio are more important than personal, social, or employment attributes like employment status for classification accuracy.

The rest of the paper is organized as follows. Section 2 provides a literature background. Section 3 discusses the six methods used. Section 4 presents the basic characteristics of the five datasets used, whereas section 5 describes the computer simulation experiments and the construction of model parameters. The results are covered and discussed in section 6. Finally, section 7 concludes the paper and outlines possible future research in this area.

BACKGROUND

One of the most commonly used data mining approaches in credit scoring research is NNs. Khashman (2009) uses NNs on an Australian dataset and finds that single-hidden layer NN outperforms double-hidden layer NN, and that a training to validation ratio of 43.5:56.5 percent is the best training scheme on the data. Baesens, Van Gestel et al.(2005) use NNs and LR on a dataset from a UK financial institution and find that the NN approach does not significantly outperform the estimated proportional hazards models. West (2000) tests five NN architectures (multilayer perceptron (MLP), mixture-of-experts (MOE), RBFNN, learning vector quantization (LVQ), and fuzzy adaptive resonance (FAR) against LDA, LR, *k*NN, kernel density estimation (KDE), and DTs on credit datasets from the University of Hamburg (Germany) and Australia using 10-fold cross-validation. The study finds that among neural architectures the 'mixture-of-experts' and RBFNN perform the best, whereas among the traditional methods LR analysis is the most accurate.

The application of SVMs in credit scoring models is more recent (Belloti & Crook, 2009). Li, Shiue, and Huang (2006) use SVM on a real world dataset from Taiwan and compares it to NN. They find that SVM surpasses traditional NN models in generalization performance and visualization. Bellotti and Crook (2009) use SVM, LR, LDA and *k*NN on a very large dataset (25000 records) from a financial institution and find that SVM is comparatively successful in classifying credit card debtors who do default, but unlike other similar models, a large number of support vectors are required to achieve the best performance.

Some researchers have used hybrid methods, and ensemble methods. Lee and Chen (2005) use a hybrid NN and multivariate adaptive regression splines (Standifird & Marshall) model and compare it to LDA, LR, NN, and MARS models on a real world housing loan dataset from a bank in China and find that hybrid NN outperforms LDA, LR, NN, and MARS. Lee and Chen (2009) use hybrid SVM, classification and regression tree (CART), MARS and grid search on a credit card dataset from a bank in China and find that the hybrid SVM has the best classification rate and the lowest Type II error in comparison with CART, MARS. Paleologo, Elisseeff and Antonini (2010) employ subbagged versions of kernel SVM, *k*NN, DTs and Adaboost on a real world dataset of IBM's Italian customers and find that subbagging, an ensemble classification technique for unbalanced datasets, improves the performance of the base classifier, and that subbagged DTs result in the best-performing classifier. Yu, Wang and Lai (2009) use individual and ensemble methods for MLR, LR, NN, RBFNN, and SVM. Their ensemble models' decisions are based on fuzzy voting and averaging, and group decision making. Three datasets are used in the study including a modified version of the Australian dataset (without missing values) and the German dataset described later in this paper. Yu, Wang and Lai (2009) find that a fuzzy group decision making (GDM) model outperforms other models on all 3 datasets. Chrzanowska, Alfaro and Witkowska (2009) use classification trees with boosting and bagging on a real world dataset from a commercial bank in Poland. They find the best performer to be an ensemble classifier using boosting with respect to accuracy and the identification of non-creditworthy borrowers. Two comparative studies (Zurada, 2007, 2010) use LR, NN, DT, memory-based reasoning (MBR), and an ensemble model using the German and SAS-1 datasets described later in this paper. Both studies find that for some cut-off points and conditions DTs perform well with respect to classification accuracy and that DTs are attractive tools for decision makers because they can generate easy to interpret if-then rules. Finally, in their preliminary computer simulation conducted on all five datasets (Tables 2-3), Zurada and Kunene (2010, 2011) describe initial findings with respect to the six methods and five datasets used in their study.

Other studies have compared expert systems and genetic programming methods. Ben-David and Frank (2009) benchmark an expert system against NN, LR, Bayes, DT, *k*NN, SVM, CT, and RBFNN using a dataset from an Israeli financial institution. They find that when a problem is treated as a regression, some machine learning models can outperform the expert systems with statistical significance, but most models do not. When the same problem is treated as a classification problem, however, no machine learning model outperforms the expert systems. Chen and Huang (2003) use an NN and genetic algorithm on data from the University of California Irvine (UCI) machine learning repository and report that using a Genetic Algorithm (GA)-based inverse classification allows creditors to suggest conditional acceptance and further explain the conditions used for rejection. Lee, Chiu, Chou and Lu (2006) use CART, MARS, LDA, LR, SVM on a real world bank credit card dataset from China and find that CART and MARS outperform traditional DA, LR, NN, and SVM with respect to accuracy on that dataset. Table 1 summarizes the previous studies on credit worthiness.

Table 1. Relevant studies on credit worthiness.

Author(s), Year	Technique(s) Used	Dataset(s) Used	Performance Measures	Findings of the Study
Hendley	*k*NN	Large mail	Minimization of	Adjusted Euclidean

and Hand (1996)	benchmarks: LR, DT, regression, decision graphs	order company dataset	bad risk rate among those accepted	distance *k*NN outperforms other models on an unbalanced dataset
West (2000)	LR, LDA, KNN Kernel Density (KD), RT, 5 NN models. (MOE, RBF, MLP, LVQ, FAR)	Australian, German	Classification Accuracy, Cost of Error. 0.5 = cut off point	Australian Best Models: MOE, RBF, MLP, LR, LDA, KNN. Worst Models: LVQ, FAR, KD, RT. German Best Models: MOE, RBF, MLP, LR. Worst Models: LVQ, FAR, LDA, KNN, KD, RT. Nonparametric models maybe better suited for large datasets
Chen and Huang (2003)	NNs with Genetic Algorithms (GAs) for inverse classification. Benchmarks: LDA, CART	Australian	Classification Accuracy. 0.5 = cut off point	LDA and CART models more accurate at classifying good loans; NN more accurate classifying bad loans
Lee, Chiu, Chou and Lu (2006)	CART, MARS. Benchmarks: LDA, LR, NN, SVM	One dataset from a Taipei bank	Average classification rate Type I Error Type II Error	CART and MARS outperform LDA, LR, NN, and SVM
Baesens, Van Gestel et al.(2005)	Survival Analysis: NN, Proportional Hazards. Benchmarks: LR	One dataset from a UK financial institution	Classification accuracy, uses confusion matrix	NN did not significantly outperform proportional hazards models.
Khashman (2009)	NNs compares single hidden layer (SHNN) vs. double hidden layer NN (DHNN)	Australian	Accuracy	The SHNN outperforms the DHNN
Li, Shiue, and Huang (2006)	SVM. Benchmarks: MLP NN.	A dataset from Taiwan bank	Classification accuracy Type I error Type II error	SVM outperforms MLP
Lee and Chen (2005)	Two-stage hybrid MARS/NN model Benchmarks: LDA, LR, BPN, MARS	A dataset from a Taiwan bank	Classification accuracy Type I and Type II Errors, Expected Misclassification Costs	Hybrid model outperforms LDA, LR, MARS and BPN with respect to (wrt.) expected misclassification costs.
Paleologo, Elisseeff and Antonini (2010)	Subbagging (ensemble) classification techniques applied to: Linear SVM, Poly SVM, NN, J48 DT, RBF, SVM	Dataset from IBM's Global Finance Italian clients	AUC. Probability of a customer default using posterior probabilities, also used to identify cutoffs	Subbagging on DTs, linear SVM and RBF are by far the best.
Yu, Wang and Lai (2009)	Intelligent-agent-based fuzzy group decision making (GDM) model using NN and SVM agents. Benchmarks: RA, LR	England dataset (Thomas, 2002), UCI Japanese Credit card Data, UCI German.	Accuracy Type I Error, Type II Error AUC (specificity, sensitivity)	Fuzzy GDM outperforms Individual (LRA, RA, NN, SVMR), Ensemble (SVMR, NN)
Bellotti and Crook	SVM. Benchmarks:	A credit card dataset from	AUC	SVMs comparatively successful. SVM can

(2009)	LRA, LR, *k*NN	an unidentified "major financial institution"		also be used for feature selection.
(Zurada, 2007)	DT (entropy, chi-squared, Gini)	Unidentified	Classification Accuracy; cutoffs at 0.3; 0.5; 0.7	Differences insignificant, but chi-squared and entropy generate the simpler trees.
(Zurada, 2010)	LR, NN, RBFNN, SVM CBR, DTs.	UCI German	Accuracy AUC.	DT models classify better than other models
Ben-David and Frank (2009)	A ''mind crafted" credit scoring expert system (ES) is compared with dozens of machine learning models (MLM).	A dataset from a leading Israeli financial institution	Accuracy (hit ratio, Cohen's Kappa, mean absolute error -regression)	Classification experiment: no MLM had statistically significant advantage over ES wrt. hit ratio, Kappa statistic. 7 MLMs had such advantage in regression case
(Chrzanowska et al., 2009)	Classification Trees, with Adaboost, Bagging	A dataset from a Polish financial institution	Specificity Sensitivity Average misclassification rate	Ensemble classifier constructed using boosting method, D1 – single classification tree based on QUEST algorithm best models
This Study	LR, NN, RBFNN, SVM, *k*NN, DT	Australian (UCI) German (UCI) SAS-1 SAS-2 Farmer	Classification Accuracy at 0.5 cutoff AUC 0.5 cutoff AUC Global	German: SVM best at 0.5 cutoff overall classification. Observations about Datasets: Database size (largeness) does not necessarily improve performance. Having only real numeric attributes decreases dataset performance. SVM better candidate on balanced datasets (global performance) NN, DT better candidates on unbalanced datasets with missing data, but *k*NN does better on unbalanced dataset if there's no missing data DTs better candidates for predicting bad loan at 0.5 cutoff Financial attributes like the debt-ratio more important than demographic, social, personal attributes like employment status

Although relatively few articles have been published in *Journal of International Technology and Information Management* on credit worthiness, there are a few studies on data mining/knowledge discovery techniques in both similar and different domains. For example, Krishnamurthi (2007) applied an unsupervised learning hierarchical clustering technique to find patterns in a small credit card database which contained data about 45 individuals only. The author segmented the customers into three clusters and found delinquency patterns in each cluster. Cluster one was the safest segment as it reflected low risk. It showed that matured adults with high levels of education, longer job tenure, and who paid their balances in full were less likely to default and be delinquent. Clusters two and three had episodes of delinquency and contained risky customers.

In another study Kumar et al. (2011) used hybrid machine learning techniques based on GA and time series analysis (TSA) to investigate data for 259 days trading values for two companies from the Indian stock market. The authors achieved about 99% accuracy in predicting the next day stock market values.

This paper evaluates the performance of six methods on five different datasets to offer more contextualized understanding of the compatibility of methods and datasets. Though the methods considered in this study have been applied to credit-scoring models in the prior studies, our study offers a richer and contextualized interpretation of the application of these methods by evaluating all six models on five different real world data sets whose characteristics vary with respect to: the type and number of variables, the distribution of *bad credit* and *good credit* samples in the data, the extent of missing values, the number of samples, and country of data collection (Tables 2 and 3). The datasets used in this study are chosen because many of them are publicly available and have been used in other studies so the results of this study may be compared with past and hopefully future results of the same or similar datasets. Except the benchmarking model LR, most of the models in this study have been found to show promise in a credit worthiness context (Belloti & Crook, 2009; Henley & Hand, 1996; West, 2000).

DESCRIPTION OF SELECTED MODELS USED IN THE STUDY

This study uses six computational intelligence models. These are logistic regression (LR), neural networks (NN), decision trees (DT), radial basis function neural networks (RBFNN), support vector machines (SVM), and *k*-nearest neighbor (*k*-NN). The first three models are very well-known and have been successfully used for classification problems in many existing studies (Yuan, Li, Guan, & Xu, 2010). Examples include a standard feed-forward NN with back-propagation and a landmark C4.5 algorithm with entropy reduction for DTs (Mitchell, 1997; Quinlan, 1987, 1993). NNs encode knowledge they learn in weights linking neurons, whereas DTs store knowledge in easy to understand if-then rules. NNs have proven to be very effective classifiers in many domains as they use all input variables together to build nonlinear boundaries to separate data. However, it may be difficult to extract if-then rules from their weights. On the other hand, DTs generate easy to interpret rules, but create linear partitions to separate data using one variable at a time. Consequently, we only provide the fundamental properties of the three remaining models used in our study. These are RBFNN, SVM, and *k*-NN.

Radial Basis Function Neural Network

An RBFNN consists of two layers, a hidden layer and an output layer. It differs from a feed-forward NN with back-propagation in the way the neurons in the hidden layer perform computations (Mitchell, 1997). Each neuron in a hidden layer represents a point in input space and its output for a given training pattern depends on the distance between its point and the pattern. The closer these two points are, the stronger the activation. The RFBNN uses Gaussian activation functions u_j whose width may be different for each neuron. The output u_j of the *j*th hidden neuron is given by $u_j = \exp\left[-\frac{(\mathbf{x}-\mu_j)^T(\mathbf{x}-\mu_j)}{2\sigma_j^2}\right]$, where $j = 1, 2, \ldots, m$, and m is the number of hidden neurons, **x** is the input pattern vector, $\boldsymbol{\mu_j}$ is its input weight vector (the center of the Gaussian for node *j*), and σ_j^2 is the normalization parameter, such that $0 \le u_j \le 1$ (the closer the input to the center of the Gaussian, the larger the response of the neuron).

The output layer forms a linear combination from the outputs of neurons in the hidden layer of the form $y_j = \mathrm{w}_j{}^{\mathrm{T}}\mathrm{u}$, $j = 1, 2, \ldots, l$, where l is the number of neurons in the output layer, y_j is the output from the *j*th neuron in the output layer, $\mathbf{w}_j$ is the weight vector for this layer, and $\mathbf{u}$ is the vector of outputs from the hidden layer.

A network learns two sets of parameters. First, it learns the centers and width of the Gaussian functions by employing the *c*-means clustering algorithm and then it uses the least mean square error algorithm to learn the weights used to form the linear combination of the outputs obtained from the hidden layer. As the first set of parameters can be obtained independently of the second set, RFBNN learns almost instantly if the number of hidden units is much smaller than the number of training patterns. Unlike a feed-forward NN with back-propagation, the RBFNN, however, cannot be trained to disregard irrelevant variables because it gives them the same weight in distance calculations.

Support Vector Machines

SVM, originally developed by Vapnik (1998), is a method that represents a blend of linear modeling and instance-based learning to implement nonlinear class boundaries. This method chooses several critical boundary patterns, called support vectors, for each class (*bad loan* and *good loan* of the output variable) and creates a linear discriminant function that separates them as widely as possible by applying a linear, quadratic, cubic or higher-order polynomial term decision boundaries. A hyperplane that gives the greatest separation between the classes is called the maximum margin hyperplane in the form of $x = b + \sum \alpha_i y_i (\mathrm{a}(i) \cdot \mathrm{a})^n$ where *i* is support vector, y_i is the class value of training pattern $\mathbf{a}(i)$, *b* and α_i are parameters that represent the hyperplane and are determined by the learning algorithm. The vectors $\mathbf{a}$ and $\mathbf{a}(i)$ represent a test pattern and support vectors, respectively. $(\mathbf{a}(i) \cdot \mathbf{a})^n$, which computes the dot product of the test pattern with one of the support vectors and raises the result to the power n, is called a polynomial kernel. One approach to determine the optimal value for *n* is to start with a linear model (*n*=1) and then increment it by a small value until the estimated error stops to decrease. Other two common kernel functions could also be used to implement a different nonlinear mapping. These are the radial basis function kernel and the sigmoid kernel. Which kernel function generates the best results is often determined by experimentation and depends on the application at hand as well. Constrained quadratic optimization is applied to find support vectors for the pattern sets as well as parameters *b* and α_i. Compared with DTs, for example, SVMs are slow but often yield more accurate classifiers because they create subtle and complex decision boundaries.

The k-Nearest Neighbor Method

In classifying a new case, the *k*-NN approach retrieves the cases it deems sufficiently similar and uses these cases as a basis for the new case (Mitchell, 1997). The *k*-NN algorithm takes a dataset of existing cases $(\mathbf{x}, y) \in D$ and a new case, $z = (\mathbf{x}', y')$ to be classified, where each existing case in the dataset is composed of a set of variables and the new case has one value for each variable. The normalized Euclidean distance or Hamming distance D_z, between each existing case and the new case (to be classified) is computed. The *k* existing cases that have the smallest distances to the new case are the *k*-nearest neighbors to that case. Based on the target values of the *k*-nearest neighbors, each of the *k*-nearest neighbors votes on the target value for the new case. The votes are the posterior probabilities for the class dependent variable.

The new case is classified based on the majority class of its nearest neighbors. Majority voting is defined as follows: $y' = \underset{v}{\operatorname{argmax}} \sum_{(x_i, y_i) \in D_z} I(v = y_i)$ where *v* is a class label, y_i is the class label for one of the nearest neighbors, and $I(\cdot)$ is an indicator function that returns the value 1 if its argument is true and 0 otherwise.

In the majority voting approach every neighbor has the same impact on the classification. This makes the algorithm more sensitive to the choice of *k*. To reduce the influence of *k* one can weigh the impact of each nearest neighbor $\mathbf{x}_i$ according to its distance: $w_i = 1/d(\mathbf{x}', \mathbf{x}_i)^2$
As a result, training patterns that are located far away from *z* will have a smaller influence on the classification compared to those that are located closer to *z*. Using the distance-weighted voting scheme, the class label of the new case can be determined as follows:
Distance-Weighted Voting: $y' = \underset{v}{\operatorname{argmax}} \sum_{(x_i, y_i) \in D_z} w_i \times I(v = y_i)$

There are two critical choices in the *k*-NN method, namely, the distance function and the cardinality *k* of the neighborhood.

DATASETS USED IN THIS STUDY

We have chosen five datasets from different financial contexts. In some cases, the datasets also describe family, social as well as personal characteristics of the loan applicants. In one of the five datasets the names of attributes are concealed for confidentiality reasons. The five datasets differ in the following ways: number of cases, types of attributes, ratio of good to bad loans, and country of origin (three different countries). However all datasets were produced to determine the credit worthiness of customers. In nearly all five cases the datasets contain information about loan applicants that the (data collecting) financial institutions deemed to be creditworthy individuals to extend a loan to. Note: the nature of the problem is such that there would have been other applicants who did not qualify for a loan at the time of application and are therefore not included in the datasets. Although this situation does not impact the validity of our analysis, we should bear in mind that we cannot know if the excluded applicants would have paid off or defaulted on a potential loan. We are interested in assessing the amenability of the datasets to the credit-scoring task. For simplicity, we refer to this amenability as the "quality" of the datasets.

The use of multiple datasets is important in the context of our paper because the existing studies show mixed results but it is difficult to compare their results as datasets in different studies tend to be different. Our study brings the different datasets under the same simulation conditions thus allowing us to observe the effect of their idiosyncrasies.

The general features of each dataset are described below. The German and Australian datasets are publicly available at the UCI Machine Learning Repository at http://www.ics.uci.edu/~mlearn/databases/, and SAS-1 and SAS-2 datasets are derived from the HMEQ dataset. The latter resides on the SAMPSIO library which can be accessed from within SAS and SAS Enterprise Miner. Depending on the method, the values of the numeric attributes were normalized to the [-1, 1] range or to a zero mean and a unit variance. Variable and value reduction techniques are separately discussed at the end of the section on results. Below is a description of each of the datasets.

Table 2. General characteristics of the five datasets used in the study.

Dataset	Characteristics				
	Cases	Attributes	Categorical	Numeric	Target variable: B = bad loans G = good loans
Australian	690	15	9	6	B: 383 G: 307
SAS-1	5960	12	2	10	B: 1189 G: 4771
SAS-2	3364	12	2	10	B: 300 G: 3064
German	1000	20	12	8	B: 300 G: 700
Farmer	244	15	1	14	B: 65 G: 176

Table 3. General description of the datasets.

Dataset	Description
Australian	The dataset describes financial attributes of Japanese credit card customers. It is available at the UCI Machine Learning Repository. The attributes names are not revealed. Though not large in size, it is well balanced with bad loans slightly overrepresented (55% and 45% of bad loans and good loans, respectively). The dataset contains a mixture of continuous variables and nominal variables and there are some missing values. Two nominal variables take a large number of distinct values (9 and 14) and six remaining variables have only 2 or 3 distinct values. The dataset has been used in more than a dozen of studies, which include for example, Quinlan (1987, 1993) who tested improvements to the DT algorithms he had proposed as well as other researchers (Boros et al., 2000; Eggermont, Kok, & Kosters, 2004; C. L. Huang, Chen, & Wang, 2007; Luo et al., 2009).
SAS-1	The dataset describes a financial data of home improvement and debt consolidation loans. The dataset contains attributes that are continuous, and nominal (with a small number of distinct values) that describe financial, and some personal characteristics of the loan applicants like type of employment. It is an unbalanced dataset where bad loans are underrepresented by a ratio of about 1:4. The dataset contains a large number of missing values which are replaced using imputation techniques. The dataset is available from the SAS Institute, including the description of attributes. This dataset has been used in a few studies, for example (Zurada, 2007, 2010).
SAS-2	The dataset describes financial data of home improvement and debt consolidation loans. It contains attributes that are continuous, and nominal (with a small number of distinct values) that describe financial, and personal characteristics of loan applicants. It is a very unbalanced dataset with bad loans significantly underrepresented by a ratio of approximately 1:10. It is obtained from the SAS-1 dataset by removing all missing values. Though the dataset has the same variables as the SAS-1 dataset and approximately 50% of the same cases, we consider it a different dataset as the ratio of

	bad loans to good loans has changed dramatically. This dataset has been used in a few studies, for example (Zurada, 2007, 2010).
German	The dataset is obtained from a German financial institution for various loan types. It describes financial, personal, and familial information about the applicants. The dataset is unbalanced as bad loans are underrepresented (30% of bad loans and 70% of good loans). It is available at the UCI Machine Learning Repository. It contains eight numeric attributes, twelve categorical attributes, and there are no missing values. One of the nominal attributes has 10 unique values and the remaining attributes have between 2 and 5 distinct values. The names of the attributes are available. The dataset seems richer than the rest because it contains personal and demographic data that is not captured in the other datasets. The dataset has been used extensively in a number of studies, for example (Huang, Chen, Wang, 2007; Luo, Cheng, Hsieh, 2009).
Farmer	The dataset is the smallest of the five datasets and is an unbalanced dataset where bad loans are underrepresented (27% of bad loans and 73% of good loans) and the names of the attributes are available. The dataset includes one nominal variable and the rest are continuous variables that include financial ratios that describe each farm borrower's financial profile. There are no missing values. The dataset was collected from Farm Service Agency (FSA) loan officers and has been used in several studies (Barney, Graves, & Johnson, 1999; Foster, Zurada, & Barney, 2010).

MODEL PARAMETER SETTINGS AND PERFORMANCE METRICS

Weka 3.7 (www.cs.waikato.ac.nz/ml/weka/) is used in this study to perform all the computer simulations. There are multiple approaches to parameter optimization (Belloti & Crook, 2009; Kecman, 2001). In this study, for LR and DT models we used standard/default Weka settings. However, the parameters for the NN, RBFNN, SVM, and kNN models were tuned for the best performance on each corresponding dataset using the Weka CVParameterSelection meta-classifier, which implements a grid search. After finding the best possible setup, the meta-classifier then trains an instance of the base classifier with these parameters and uses it for subsequent predictions on the test sets.

More specifically:

- The LR used a quasi-Newton Method with a ridge estimator for parameter optimization (le Cessie & van Houwelingen, 1992).

- The DT generated a pruned C4.5 decision tree (Quinlan, 1987). The confidence factor that determines the amount of pruning was set to 0.2. Smaller values assigned to the confidence factor would incur more pruning.

- The standard 2-layer feed-forward NN with back-propagation was used. Momentum was set to 0.2, and the learning rate was initially set to 0.3. A decay parameter, which causes the learning rate to decrease, was enabled. This may help to stop the network from diverging from the target output as well as improve general performance. Depending on the dataset, the number of neurons in a single hidden layer varied from 9 to 23 and was computed as *a=(number of attributes including dummy attributes)/2+1*.

- The RBFNN implemented a normalized Gaussian radial basis function network. It used the *k*-means clustering algorithm to provide the basis functions and learn a logistic regression on top of that. Symmetric multivariate Gaussians were fit to the data from each cluster. The minimum standard deviation for the clusters varied between 0.4 and 1.6, and the number of clusters varied from 4 to 14 for the five datasets.

- The SVM implemented Platt's sequential minimal optimization (SMO) algorithm for training a support vector classifier (Keerthi, Shevade, Bhattacharyya, & Murthy, 2001; Platt, 1998). Depending on the dataset, the complexity parameter C and the power of the polynomial kernel was set to n=1 or n=2. Also, RBF kernel was used with γ=0.01. The grid method was used to find the optimal parameters for C, n, and γ.

- The *k*NN implemented a *k*-nearest neighbor classifier (k=10) according to the algorithm presented by Aha and Kibler (1991). The Euclidean distance measure is used to determine the similarity between the samples. The inverse normalized distance weighting method and the brute force linear search algorithm were used to search for the nearest neighbors. For each dataset, we performed several experiments for different

> values of *k* and used the normalized Euclidean distance for numeric variables and the Hamming distance for nominal variables to calculate the similarity between cases. The numeric attributes were normalized to ensure that features with larger values do not overweight features with lower values. Furthermore, to minimize the influence of *k*, we used the voting approach with weighted-distance.

Ten-fold cross-validation was applied to each of the six methods and five dataset pairings investigated in this study using a methodology as described in Witten and Frank (2005). To obtain reliable and unbiased error estimates each experiment was repeated 10 times. The performance measures of the methods and datasets were then averaged across these 10 folds and 10 runs, and a two-tailed paired *t*-test (at α=.05 and α=.01) was used to verify whether the classification performances across the models and datasets were significantly different from the baseline (LR) method and the baseline (Australian) dataset. In other words, we state hypotheses in an implicit way. For example, using LR as the benchmark one can state the following hypothesis and perform a two-tailed *t*-test: *The overall rate generated by a model (for example, NN) is significantly better/worse than the rate generated by LR*.

The LR method was used as the baseline because this traditional technique has been successfully applied to classification problems going back many years, before computational intelligence techniques emerged. The Australian dataset was chosen as the baseline as it appears to have the "best" attributes and other data characteristics and all the past models built on it consistently exhibited the highest classification performance. The parameters for the models on each dataset were optimized for the best performance.

We use the overall correct classification accuracy rates as well as the classification accuracy rates for good and bad loans (at a standard 0.5 cutoff point) to evaluate the performance of the six methods across the five datasets and the five datasets across the six methods. In other words if the target event is "detecting bad loans" and the model generates a loan default probability ≥ 0.5, the individual should not be granted a loan. We should point out that though the choice of this 0.5 cutoff point is found in a majority of existing studies, it is not always appropriate as it assumes that the costs of misclassifying a *good loan* is the same as that of misclassifying a *bad loan*. In practice this is not always the case. Thus financial institutions may choose any cutoff point within the [0, 1] range to approve or deny a loan. For instance, if the target event is "detecting bad loans" and the cost of classifying a *bad loan* as a *good loan* is 2.33 times greater than the cost of classifying a *good loan* as *bad loan*, a 0.3 cutoff point should be used. This cutoff point may be applicable in situations where banks do not require security or collateral for smaller loans. Consequently, if a model produces a probability of loan default ≥ 0.3, the customer will be denied a loan. If, however, financial institutions secure larger loans by holding collateral such as the customer's home, a more lenient cutoff point of, say, 0.7 could be applied. Therefore in practice, ROC chart(s) and the area(s) under the curve(s) are useful analytics tools, because they capture the global performance of the methods and datasets at *all* operating points within the range [0,1] as well as the performance of the methods and datasets at specific cutoff points.

A ROC chart plots a true positive rate (TPR) vs. a false positive rate (FPR) for all cutoff points within the [0,1] range. Each point on a curve represents a cutoff probability. However, the exact locations of the cutoff probabilities are difficult to pinpoint on every chart because they depend on the method and the dataset, i.e., they vary from method to method and from dataset to dataset. Points in the lower left corner and in the upper right corner represent high and low cutoff probabilities, respectively. The extreme points (1,1) and (Lenat) represent no-data rules where all cases are classified into bad or good loans, respectively. The area under the curve gives a quantitative measure of performance: the higher the classification accuracy, the further the ROC curve pushes upward and to the left. The area under the curve ranges from 50% for a worthless model, to 100% for a perfect classifier.

Tables 4 to 6 present the overall, *bad loans*, and *good loans* classification accuracy rates with their respective standard deviations at a single 0.5 probability cutoff point. The areas under the ROC curves and standard deviations are shown in Table 7. With the LR method as the baseline method we compare the six methods' rates on each of the five datasets (across the table rows). We suffix the performance rate with the superscripts ^{b,bb} and ^{w,ww} to indicate whether each one of the five other methods performs significantly better or worse (at α=0.05 and α=0.01 respectively) than the baseline method LR. The LR method is chosen as the baseline because it was very often used as the primary and the only method in early studies on creditworthiness and bankruptcy predictions. Down the table columns, we compare the performance of the five datasets for each of the six methods with the Australian dataset as the baseline. We prefix the rate with subscripts $_{b,bb}$ and $_{w,ww}$ to indicate whether each dataset is significantly better or worse (at α=0.05 and α=0.01, respectively) than the Australian (i.e. baseline) dataset in terms of classification performance. The Australian dataset is used as the baseline as it exhibits the best classification performance on all six methods compared to other datasets. Furthermore, we average the classification accuracy rates in the rows and columns by data method and by dataset, to obtain a more general insight into the performance characteristics of the methods and datasets. We also rank the methods (last row) and datasets (last column) of Tables 4-7 using the average scores.

The ROC curves in Figures 1-5 compare the global performance of the six methods for each of the five datasets, while the ROC curves in Figures 6-11 compare the five datasets for each of the six methods. All the presented ROC charts capture the performance of the methods and datasets for *bad loans*, i.e., each method's correctly classified loan defaults divided by the total number of loan defaults are plotted on the Y-axis (sensitivity). The X-axis plots good loans incorrectly classified as bad loans divided by the total number of good loans (1-specificity). We assume the detection of bad loans is more important than the detection of good loans for credit granting institutions, thus even though it would be easy to show corresponding charts for good loans, we do not do so for this study.

RESULTS AND DISCUSSION

In this section we present the results of the experiments and provide an in-depth discussion of these results. First the overall classification rates are examined and compared across the different models and the different datasets. This is followed by a more detailed look where the models and the datasets are evaluated for their classification accuracy for bad loans or good loans. Then an analysis of the areas under the curves in ROC charts is provided. Finally, we present feature reduction techniques applied to the 5 datasets and discuss their effect on the performance of the models as well as list the relevant features which were retained in each dataset.

Overall classification: the models

In this section we report on the results of applying the methods to the data. We refer here to LR, SVM, DT, RBFNN, NN and *k*NN as models rather than methods (Hevner et al., 2004; March & Smith, 1995)

Table 4 shows that, with respect to the *overall* classification accuracy rates, the NN (85.8%), RBFNN (86.2%), and *k*NN (86.3%) models significantly outperform the baseline LR (85.2%) model on the better balanced Australian dataset where bad loans are slightly overrepresented[1]. There is not, however, a significant difference between the performance of LR versus SVM (85.6%) and DT (85.6%). The average standard deviations (spreads) of the classification accuracy rates seem relatively small and amount to just 3.9%. For the unbalanced SAS-1 dataset the NN, RBFNN, SVM, and DT models classify cases significantly better than LR, whereas *k*NN is the only model which performs significantly worse than LR. For the SAS-2 dataset, SVM (93.4%) and DT (94.4%) perform significantly better than LR (92.5%), whereas the remaining three models classify cases significantly worse. The high overall classification accuracy rates on the SAS-2 dataset are due to the fact that this dataset is heavily overrepresented by good loans. For the SAS-1 and SAS-2 datasets the average spreads in the rates are very small and equal to 1.1% and 0.8%, respectively. For the German dataset LR and SVM seem to significantly outperform the four remaining models. For the small Farmer dataset only the *k*NN model appears to outperform LR, whereas SVM classifies cases significantly worse than the baseline model, and the other three remaining models are no better than the baseline. One can also see that the average spreads in the rates are quite significant (6.8%). This could be attributed to the small size of this dataset. These results are consistent with those reported by Huang, Chen and Wang

[1] Note that even if the difference between the two rates (85.8% - 85.2% = 0.6%) for the two models (NN and LR) appear to be tiny, the *t*-test will still show the statistically significant difference between the classification performance of the two models. In other words, if one model generates a consistently smaller rate that than another model (even by a small amount), it is likely that the *t*-test will show the statistically significant difference. Also note that in a formula (not shown here) for computing the *t*-value includes the variances of the rates normalized by the number of samples. The above observation applies to the results presented in Tables 4-7.

(2007) with respect to the overall classification accuracy rates at the 0.5 cutoff point for the NN, DT, and SVM models when applied to the Australian and German datasets.

The above analysis offers some mixed results. At the 0.5 cutoff point there is not a clear and sustained pattern in terms of the superiority of one model over another that could be generalized and tied to particular features of the datasets with perhaps one exception. DTs perform significantly better than the other models on SAS-1 and SAS-2 datasets. These datasets grossly underrepresent the number of *bad loans*. The last row on Table 4 shows the averages of the overall classification accuracy rates for each model across the five datasets and suggests that the differences between the models are small. NN (83.7%) seems to perform best followed by DT (83.6%), RBFNN (83.1%), SVM (83.0%), LR (82.9), and *k*NN (82.5%). From a practitioner point of view, this may be encouraging because it suggests in this case that a choice to use DTs for their utility as a readily interpretable model isn't necessarily at the expense of foregoing large degrees of classification accuracy relative to alternative models.

Overall classification: the datasets

The analysis down the columns (of Table 4) enables us to assess the quality of each dataset used to build the models. The SAS-2 dataset appears to have the most favorable characteristics, as the six models built on it have the highest mean overall classification accuracy rate (92.8%). This may be largely due to the fact that this dataset is heavily predominated by *good loans*, i.e., at a ratio of 10:1, and they classify *good loans* almost perfectly well. For the Australian, SAS-1, Farmer, and German datasets the six models exhibit the average classification rates of 85.8%, 84.7%, 77.5%, and 74.9%.

Table 4. The average overall correct classification accuracy rates [%] and standard deviations at a 0.5 probability cutpoint.

	LR	NN	RBFNN	SVM	*k*NN	DT	Avg	AvgRank
Australian	85.2 4.0	85.8^{b} 3.8	86.2^{bb} 4.1	85.6 3.7	86.3^{bb} 3.8	85.6 3.7	85.8 3.9	2
SAS-1	${}_{ww}83.6$ 1.0	${}_{bb}86.9^{bb}$ 1.3	${}_{ww}84.6^{bb}$ 1.1	${}_{w}84.8^{bb}$ 1.0	${}_{ww}79.1^{ww}$ 1.3	${}_{bb}88.9^{bb}$ 1.0	84.7 1.1	3
SAS-2	${}_{bb}92.5$ 0.7	${}_{bb}92.1^{ww}$ 0.6	${}_{bb}92.2^{ww}$ 1.1	${}_{bb}93.4^{bb}$ 0.7	${}_{bb}92.4^{w}$ 0.5	${}_{bb}94.4^{bb}$ 1.0	92.8 0.8	1
German	${}_{ww}75.8$ 3.8	${}_{ww}75.4^{w}$ 3.8	${}_{ww}75.0^{w}$ 3.9	${}_{ww}75.9$ 3.6	${}_{ww}74.6^{ww}$ 3.4	${}_{ww}72.9^{ww}$ 4.0	74.9 3.8	5
Farmer	${}_{ww}77.2$ 7.1	${}_{ww}78.3$ 6.0	${}_{ww}77.6$ 6.0	${}_{ww}75.2^{ww}$ 8.3	${}_{ww}80.2^{bb}$ 6.5	${}_{ww}76.2$ 7.1	77.5 6.8	4
Average	82.9 3.3	83.7 3.1	83.1 3.2	83.0 3.5	82.5 3.1	83.6 3.4		
AvgRank	5	1	3	4	6	2		

Classification of bad loans: the models and datasets

Similar analyses can be undertaken for the classification accuracy rates for bad loans (Table 5) and good loans (Table 6) from the six models on each of the five datasets at a 0.5 cutoff point. For bad loans, Table 5 shows that all but one model, the SVM (80.0%), outperform LR (84.3%), with the RBFNN and DT models classifying bad loans the best with 89.0% and 87.3% classification rates on the Australian dataset. For the SAS-1 dataset all five models are better than the LR model (30.4%), with NN (59.0%) and DT (54.8%) as the best two. With respect to the SAS-2 dataset DT (47.3%), SVM (30.5%), and RBFNN (30.5%) appear to do better than the LR model (22.7%). On the remaining datasets, namely, German and Farmer datasets, NN and SVM obtain much better classification rates than the other four models. Ranking the six models with respect to the average classification rates of bad loans on the five datasets, one finds that DT (54.9%) stands out, followed by NN, SVM, RBFNN, LR, and *k*NN (44.8%) in this order. Table 5 also shows that the NN (14.2%), *k*NN (14.5%), and LR (22.7%) models built on the very unbalanced SAS-2 dataset classify bad loans very poorly.

Table 5. The average correct classification accuracy rates [%] and standard deviations at a 0.5 probability cutpoint: bad loans.

	LR	NN	RBFNN	SVM	*k*NN	DT	Avg	AvgRank
Australian	84.3 5.4	86.8^{bb} 5.0	89.0^{bb} 4.5	80.0^{ww} 5.3	88.3^{bb} 4.6	87.3^{bb} 5.2	86.0 5.0	1
SAS-1	$_{ww}30.4$ 4.0	$_{ww}59.0^{bb}$ 4.8	$_{ww}34.2^{bb}$ 4.4	$_{ww}34.6^{bb}$ 4.3	$_{ww}33.4^{bb}$ 4.6	$_{ww}54.8^{bb}$ 4.5	41.1 4.4	4
SAS-2	$_{ww}22.7$ 6.3	$_{ww}14.2^{ww}$ 6.2	$_{ww}30.5^{bb}$ 8.3	$_{ww}30.5^{bb}$ 7.4	$_{ww}14.5^{ww}$ 5.9	$_{ww}47.3^{bb}$ 9.5	26.6 7.3	5
German	$_{ww}48.3$ 8.2	$_{ww}49.7^{bb}$ 8.3	$_{ww}46.1^{w}$ 8.9	$_{ww}47.2^{w}$ 8.1	$_{ww}41.5^{ww}$ 7.9	$_{ww}44.2^{w}$ ^{w} 9.4	46.2 8.5	2
Farmer	$_{ww}44.9$ 18.5	$_{ww}34.7^{ww}$ 17.1	$_{ww}38.1^{ww}$ 19.4	$_{ww}48.5^{b}$ 18.7	$_{ww}46.4$ 17.7	$_{ww}40.9$ 19.5	42.3 18.5	3
Average	46.1 8.5	48.9 8.3	47.6 9.1	48.2 8.8	44.8 8.1	54.9 9.6		
AvgRank	6	2	5	4	3	1		

Rankings of the datasets with respect to the average classification rates of bad loans show the balanced Australian dataset stands out (86.0%) followed by a very distant German dataset (46.2%). The SAS-2 (26.6%) dataset is the worst. It appears that as the proportion of bad loans decreases, so follows the average classification accuracy of bad loans.

Classification of good loans: the models and datasets

Table 6 depicts the classification rates for good loans. The differences between the classification accuracy rates for the six models are tiny across all five datasets. For good loans SVM seems to perform the best (92.6%), followed by LR (92.0%), NN (91.9), RBFNN (91.7), *k*NN (91.3%), and DT (91.0%). And, as expected, the datasets dominated by good loans exhibit an excellent capacity to correctly classify good loans: SAS-2 (99.3%), SAS-1 (95.5%), and Farmer (91.1%). The relatively better balanced German and Australian datasets fair worse at 87.2% and 85.6% respectively. We leave the rest of the analysis to the interested readers.

When one looks at the performance of the six methods on one dataset at a time as shown in Tables 4 to 6, it is clear that it is difficult to categorically conclude or to determine which model is best and to generalize the results obtained at a standard operating cutoff point of 0.5. No one model clearly dominates the others. The quality of the models depends very much on the characteristics of the dataset such as the ratio of good loans to bad loans, the number of samples, the number and type of attributes, as described in Section 4. On the other hand ROC curves can testify to the global efficiency of a model at all operating points. Table 7 below shows a comparison of the six models for each of the five datasets using the areas under the ROC curves. Table 7 can also be analyzed in conjunction with the ROC charts presented in Figures 1-11.

Table 6. The average correct classification accuracy rates [%] and standard deviations at a 0.5 probability cutpoint: good loans.

	LR	NN	RBFNN	SVM	*k*NN	DT	Avg	AvgRank
Australian	86.4 5.6	84.5^{ww} 5.8	82.9^{ww} 6.9	92.5^{bb} 4.2	83.9^{ww} 6.4	83.5^{ww} 5.8	85.6 5.8	5
SAS-1	$_{bb}96.9$ 0.8	$_{bb}93.8^{ww}$ 1.1	$_{bb}97.1^{bb}$ 0.8	$_{bb}97.3^{bb}$ 0.8	$_{bb}90.5^{ww}$ 1.4	$_{bb}97.4^{bb}$ 0.8	95.5 1.0	2
SAS-2	$_{bb}99.4$ 0.5	$_{bb}99.7^{bb}$ 0.4	$_{bb}98.2^{ww}$ 0.9	$_{bb}99.6^{bb}$ 0.4	$_{bb}100.0^{bb}$ 0.0	$_{bb}99.0^{ww}$ 0.7	99.3 0.5	1
German	$_{b}87.5$ 4.3	$_{bb}86.4^{ww}$ 4.2	$_{bb}87.3$ 4.1	$_{ww}88.2^{bb}$ 4.0	$_{bb}88.7^{bb}$ 3.8	$_{bb}85.1^{ww}$ 4.7	87.2 4.2	4
Farmer	$_{bb}89.7$ 7.8	$_{bb}95.2^{bb}$ 5.2	$_{bb}92.9^{bb}$ 6.2	$_{ww}85.6^{ww}$ 8.7	$_{bb}93.3^{bb}$ 5.9	$_{bb}89.9$ 7.1	91.1 6.8	3
Average	92.0 3.8	91.9 3.3	91.7 3.8	92.6 3.6	91.3 3.5	91.0 3.8		
AvgRank	1	3	4	2	5	6		

ROC charts: the models

Table 7, constructed similarly to Table 4 or 6 with the six the models (across the rows) and five datasets (down the columns) shows the average areas under the ROC as a percentage and their respective standard deviations. For the Australian dataset the range in the plotted areas under the

ROC curves is between 88.2% and 92.1%, with the SVM model performs significantly better (92.1%) and the DT model significantly worse (88.2%) than the benchmark LR model (91.1%), whereas the performance of the other three models is comparable to the LR model. Figure 1 confirms the fact that the global classification accuracy rates of all the six models are excellent and that the differences in the models' performances are slight on the balanced Australian dataset. For the SAS-1 dataset all five models outperform the LR model (79.4%) and the range of areas under the ROC curves is [79.4%, 86.3%]. The NN, DT, and *k*NN models, in this order, exhibit the best overall performance, whereas the SVM, LR and RBFNN models appear to be the worst. Figure 2 provides more insight into the performance differences between the models (on the SAS-1 dataset); that is, while NN and DT appear to do better at higher operating points, *k*NN outperforms all other methods at lower cutoffs. For the highly unbalanced SAS-2 dataset the differences between the models' performances are also substantial with the range of areas under ROC curves between 75.7% (DT) and 94.2% (*k*NN). The *k*NN model and RBFNN (80.5%) perform significantly better than LR (78.7%), whereas DT is significantly worse than LR (Figure 3). However, DT and SVM tend to perform better than other models at higher operating points. For the richer German dataset only SVM (79.4%) significantly outperforms LR (79.1%) at α=0.05, whereas RBFNN (77.5%), *k*NN (75.9%), and DT (65.1%) are significantly worse at α=0.01. For the Farmer dataset, which is smaller, unbalanced, and contains mainly continuous (real) attributes, DT (59.6%) and RBFNN (71.8%) are significantly worse than LR (73.5%), while the other three models are comparable to LR. This is also evident from Figure 5. The last row in Table 7 shows the averages areas under ROC curves for each model on each of the five datasets. Compared to other models, the *k*NN models (83.2%) stand out somewhat mainly due to their excellent performance on the SAS-2 dataset, whereas DTs are noticeably inferior. The nuances evident from this analysis can contribute in guiding practitioners, faced with the realities of their own data quality, in their selection of the method most likely to perform best.

Table 7. The average areas under ROC charts [%] and standard deviations.

	LR	NN	RBFNN	SVM	*k*NN	DT	Avg	AvgRank
Australian	91.1 3.6	91.4 3.2	91.4 3.6	92.1^{bb} 3.2	91.2 3.4	88.2^{ww} 4.4	90.9 3.6	1
SAS-1	$_{ww}$79.4 2.4	$_{ww}$86.3^{bb} 2.0	$_{ww}$80.0^{bb} 2.4	$_{ww}$81.0^{bb} 2.3	$_{ww}$82.6^{bb} 1.8	$_{ww}$84.4^{bb} 2.5	82.3 2.2	2
SAS-2	$_{ww}$78.7 4.6	$_{ww}$78.0 4.2	$_{ww}$80.5^{bb} 4.4	$_{ww}$78.0 4.9	$_{bb}$94.2^{bb} 1.6	$_{ww}$75.7^{ww} 5.9	80.9 4.3	3
German	$_{ww}$79.1 4.6	$_{ww}$79.1 4.3	$_{ww}$77.5^{ww} 4.7	$_{ww}$79.4^{b} 4.3	$_{ww}$75.9^{ww} 4.7	$_{ww}$65.1^{ww} 6.3	76.0 4.8	4
Farmer	$_{ww}$73.5 11.7	$_{ww}$74.0 11.8	$_{ww}$71.8^{w} 11.2	$_{ww}$74.3 11.5	$_{ww}$72.0 11.4	$_{ww}$59.6^{ww} 13.7	70.9 11.9	5
Average	80.4 5.4	81.8 5.1	80.2 5.3	81.0 5.2	83.2 4.6	74.6 6.6		
AvgRank	4	2	5	3	1	6		

Figure 1. The ROC curves for the Australian dataset for the 6 methods.

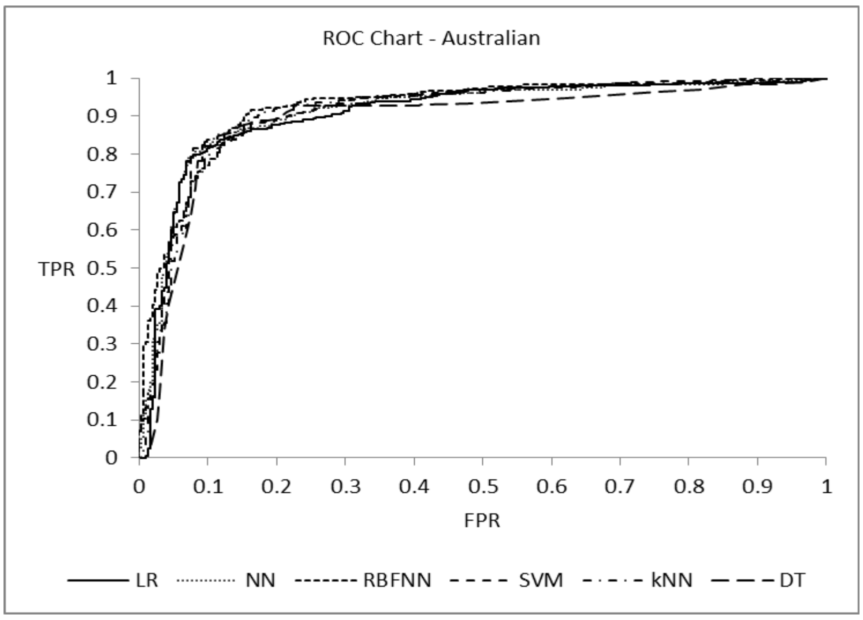


Figure 2. The ROC curves for the SAS-1 dataset for the 6 methods

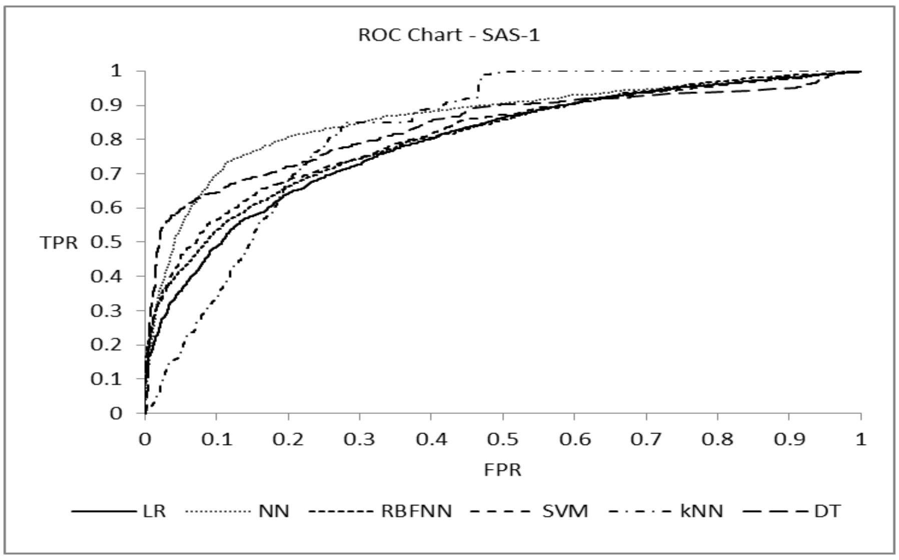


Figure 3. The ROC curves for the SAS-2 dataset for the 6 methods

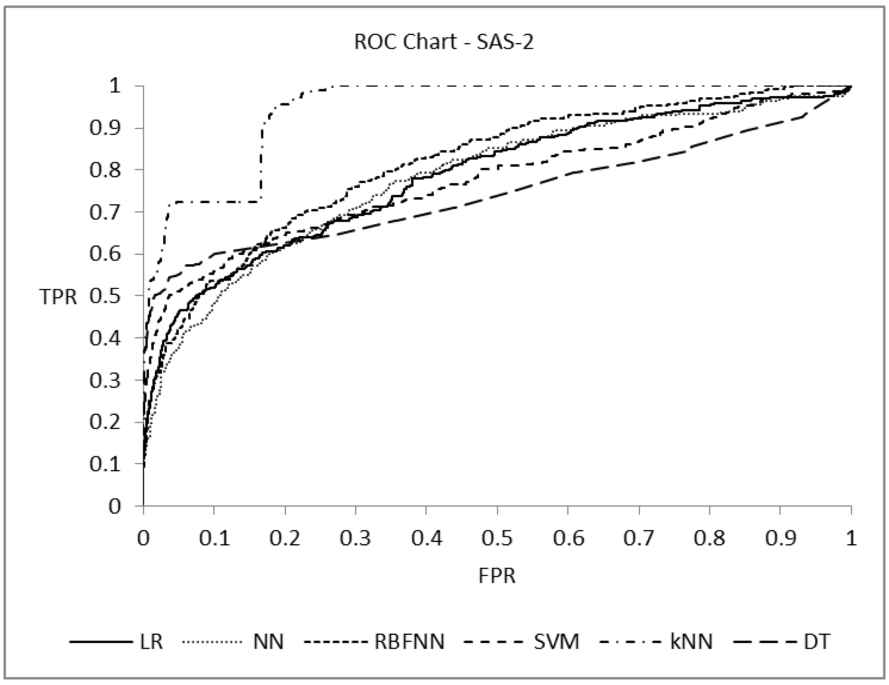


Figure 4. The ROC curves for the German dataset for the 6 methods.

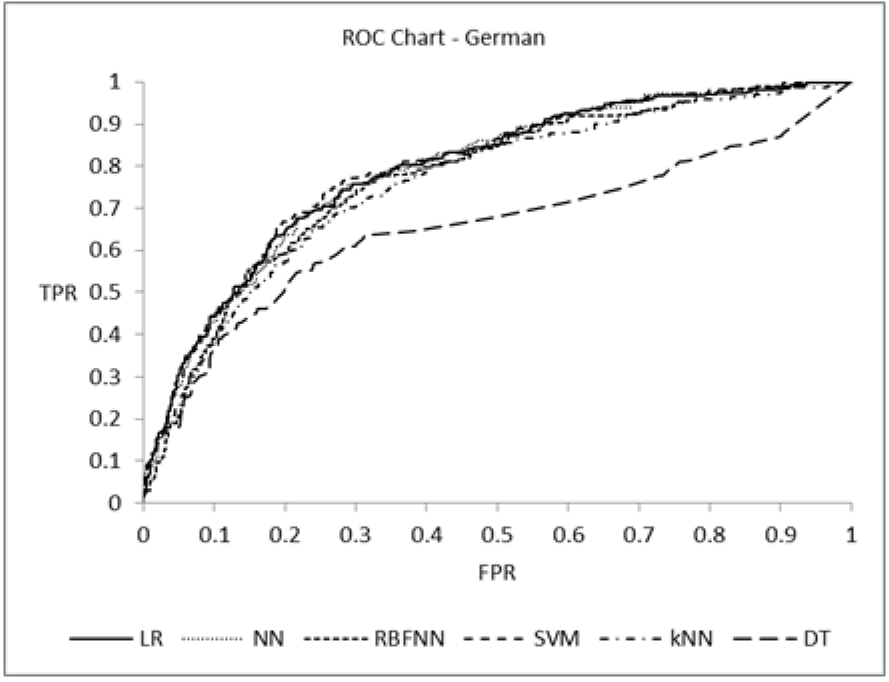
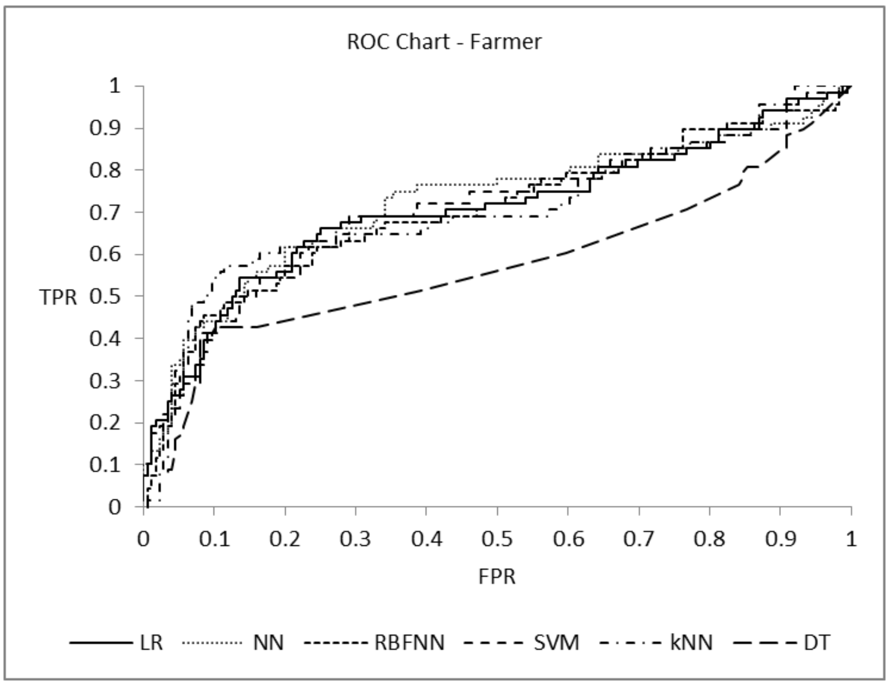


Figure 5. The ROC curves for the Farmer dataset for the 6 methods.



ROC charts: the datasets

With Table 7 and Figures 6-11 one can draw conclusions about the global quality of the datasets on which the models were built. When one analyzes the rates down the columns of Table 7, it is evident that the LR, NN, RBFNN, SVM, and DT models built on the Australian dataset perform significantly better than the models constructed on the four remaining datasets with the only exception being the *k*NN model built on the SAS-2 dataset. It is also evident that all the six models constructed on the Farmer dataset, the smallest dataset in our study, perform much worse than the models built on SAS-1, SAS-2, and German datasets. For the latter three datasets no consistent pattern of the models' performances is evident. For example, the LR, RBFNN, and SVM models built on the three datasets have roughly the same performance, whereas NN does very well on the SAS-1 dataset. For DTs, their performance depends very much on the quality of the datasets, i.e. performance gradually declines with each of the ranked datasets in our study.

The last column on Table 7 shows an ordered ranking of the datasets with respect to their quality: Australian (90.9%), SAS-1 (82.3), SAS-2 (80.9%), German (76.0%), and Farmer (70.9%). Figures 6-11 generally confirm these observations, even though as the curves intersect they can be more difficult to interpret. Figures 6 through 9 show that the LR, NN, RBFNN, and SVM models created on the balanced Australian dataset are generally the best models, whereas when built on the smaller, less balanced and the continuous attribute dominated Farmer dataset the same models are the worst. On the other hand, the differences between these same models when built on the three remaining datasets are less striking. Similar analysis of Figures 10 and 11, however, shows that the differences in global performances of the *k*NN and DT models are very big across all five datasets; the DT model is especially poor on the (Farmer) dataset containing real values. Finally, the last column in Table 7 displays the average rates over the six models for each dataset (from best to worst): Australian (90.9%), SAS-1 (82.3%), SAS-2 (80.9%), German (76.0%), and Farmer (70.9%).

Attribute reduction issues

Attribute reduction and variable worth sheds some insight on the domain variables most pertinent to predictive accuracy of the generated models. To ascertain the worth of variables we conducted attribute reduction in all five datasets. We selected two common variable reduction techniques from Weka. The first technique, CfsSubsetEval with BestFirst search method, evaluates the worth of a subset of attributes by considering the individual predictive ability of each feature along with the degree of redundancy between them. Subsets of features that are highly correlated with the class while having low inter-correlation are preferred. The BestFirst method searches the space of attribute subsets by greedy hill climbing augmented with a backtracking facility (Hall, 1998). The second technique, InfoGainAttributeEval with the Ranker search method evaluates the worth of an attribute by measuring the information gain with respect to the class. The Ranker method ranks attributes by their individual evaluations. For attribute reduction we used the same technique as for the models' building and testing, i.e., 10-fold cross-validation and repeated it 10 times. The attributes which occurred most often in the folds were selected and labeled as very significant, and the attributes which occurred less often were labeled as significant. Both the CfsSubsetEval and InfoGainAttributeEval techniques were in agreement and consistently identified the same relevant set of the attributes. These relevant attributes are shown in Table 8.

In general the attribute reduction had mixed effects on improving the overall classification accuracy rates, the rates for *bad loans* and *good loans*, as well as the global performances of the models (areas under ROC curves) in the 6 models and 5 datasets. We will first comment on the average overall rates. The rates for the LR, NN, and RBFNN models improved by about 0.5%, the rates for SVM and DTs were approximately the same, while the rates for *k*NN declined by 2%. The rates for the Australian, SAS-2 datasets remained approximately the same, while the rates for SAS-1 dataset decreased by 1% and the average rates for the German dataset improved by about 0.5%. We observed some improvements in the AUC for some models and some datasets, but these happened due to the improvements in the classification rates of good loans. However, the detection rates for bad loans did not improve, except in a few isolated cases. As detecting bad loans is more important, we decided to present the results from computer simulation for the datasets with the full set of attributes.

Variable reduction sheds some interesting insight on variable retention issues in the credit scoring domain. It appears that for the four datasets (one has hidden attributes) the financial characteristics describing customers are much more relevant than the personal, social, or employment ones (Table 8). These findings may be important for future data collection efforts by both researchers and practitioners.

Table 8. The description of the relevant and irrelevant input variables in the 4 datasets. The Australian data set in not shown as it has hidden attributes.

	Relevance of attributes		
Datasets	Very significant	Significant	Insignificant
German	- Checking account balance - Length of loan [in months] - Credit history - Savings account balance	- Reason for loan request - Credit amount - Time at present employment - Marital status & gender - Collateral property for loan - Age of applicant [in years] - Other installment loans - Rent/own a house - Foreign worker	- Debt as % of disposable income - Co-applicant or guarantor for a loan? - Years at current address - Number of accounts at this bank - Employment status - Number of dependents - Has a telephone?
SAS-1	- Amount of the loan requested - Number of major derogatory credit reports - Number of delinquent payments - Age (in months) of oldest trade line - Debt-to-income ratio - Years at present job	- Value of current property - Number of recent credit inquires - Number of trade (credit) lines	- Amount due on existing mortgage - Reason for loan: debt consolidation or home improvement - Six occupational categories
SAS-2	- Value of current property - Number of major derogatory credit reports - Number of delinquent payments - Age (in months) of oldest trade line - Number of trade (credit) lines - Debt-to-income ratio	- Amount of the loan requested - Years at present job - Number of recent credit inquires	- Amount due on existing mortgage - Reason for loan: debt consolidation or home improvement - Six occupational categories
Farmer	- Missed/delinquent payment(s) 2 years before default resulted in debt restructuring - Missed/delinquent payment(s) 1 year before		- Debt-to-equity = Total debts/(Total assets - Total debt) - Return on farm assets = (Total cash farm income from operations - Operating

	default resulted in debt restructuring - Debt-to-income ratio		expenses - Family living expenses)/Beginning total farm assets - Return on equity = (Total cash farm income - operating expenses - interest expense - family living expenses)/(Total assets - Total debt) - Operating profit margin = (Total farm income - actual operating expenses - family living expenses)/Total farm income - Projected debt repayment = (Total debt and interest payments due/(Projected total cash farm income + Non-farm income) - Debt repayment ratio = Total debt and interest payments due/(Total cash farm income + Non-farm income) - Asset turnover = Total cash farm income/Beginning total farm assets - Operating expense = Total operating expenses/Total farm income - Interest expense = Total actual interest expense paid/Total farm income

Figure 6. The ROC curves for LR for the 5 datasets.

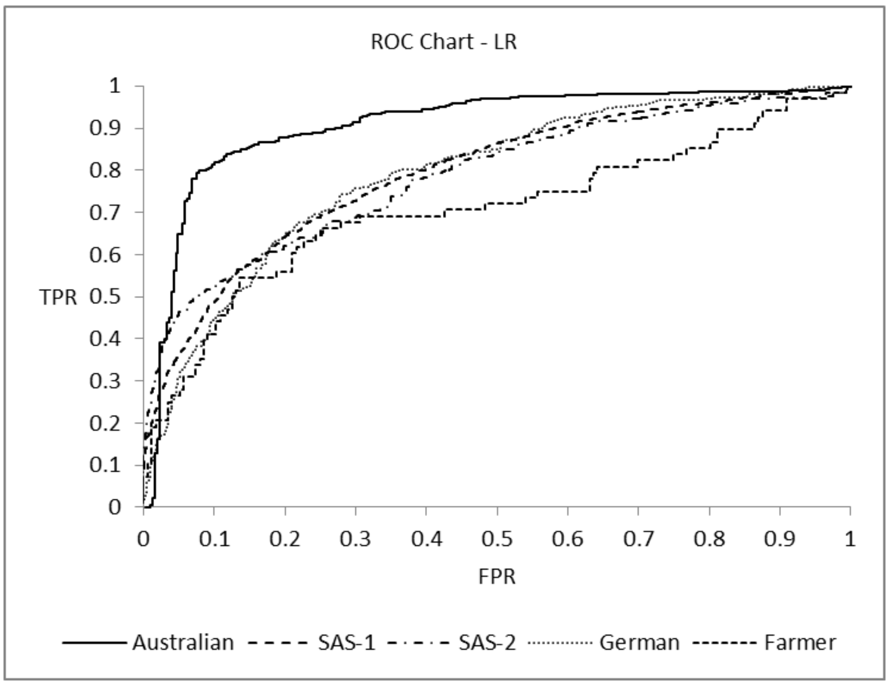


Figure 7. The ROC curves for NN for the 5 datasets

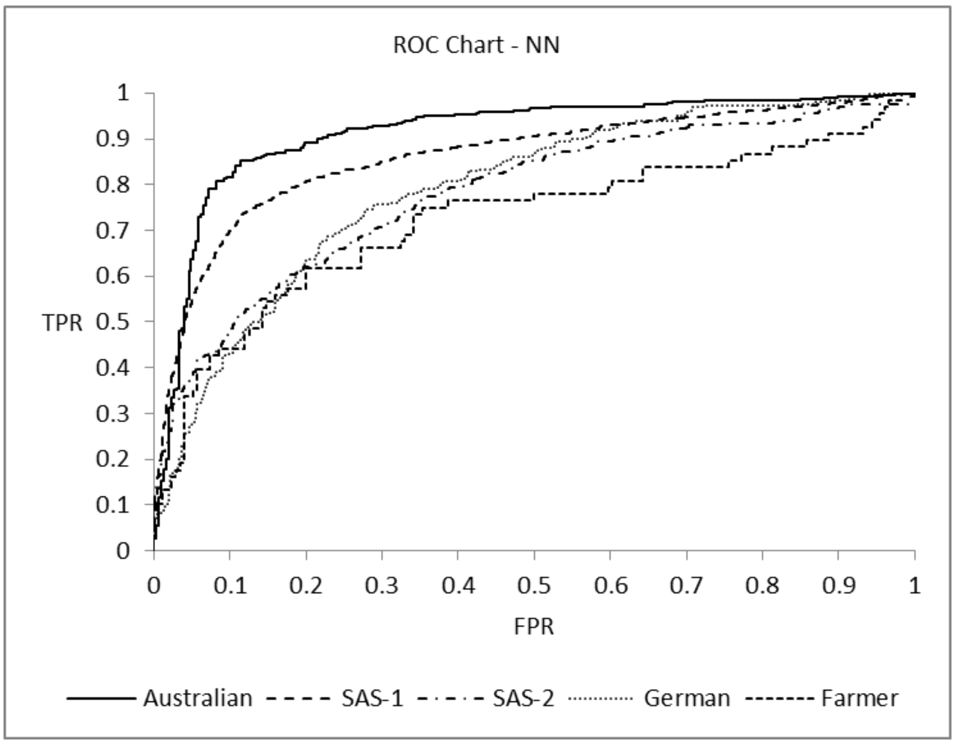


Figure 8. The ROC curves for RBFNN for the 5 datasets.

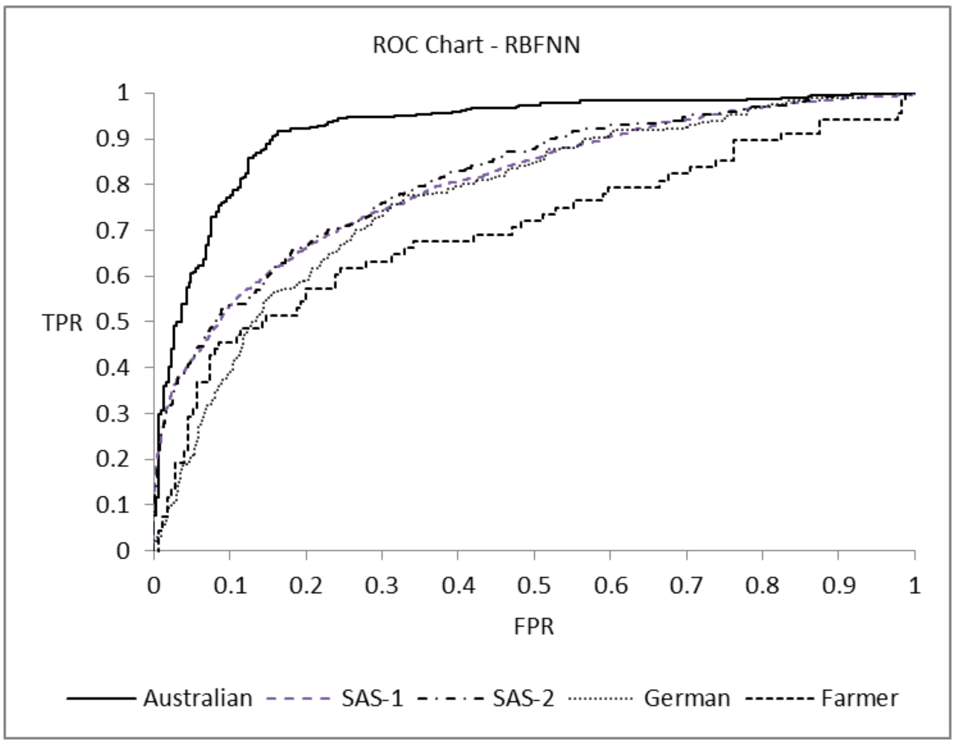


Figure 9. The ROC curves for SVM for the 5 datasets.

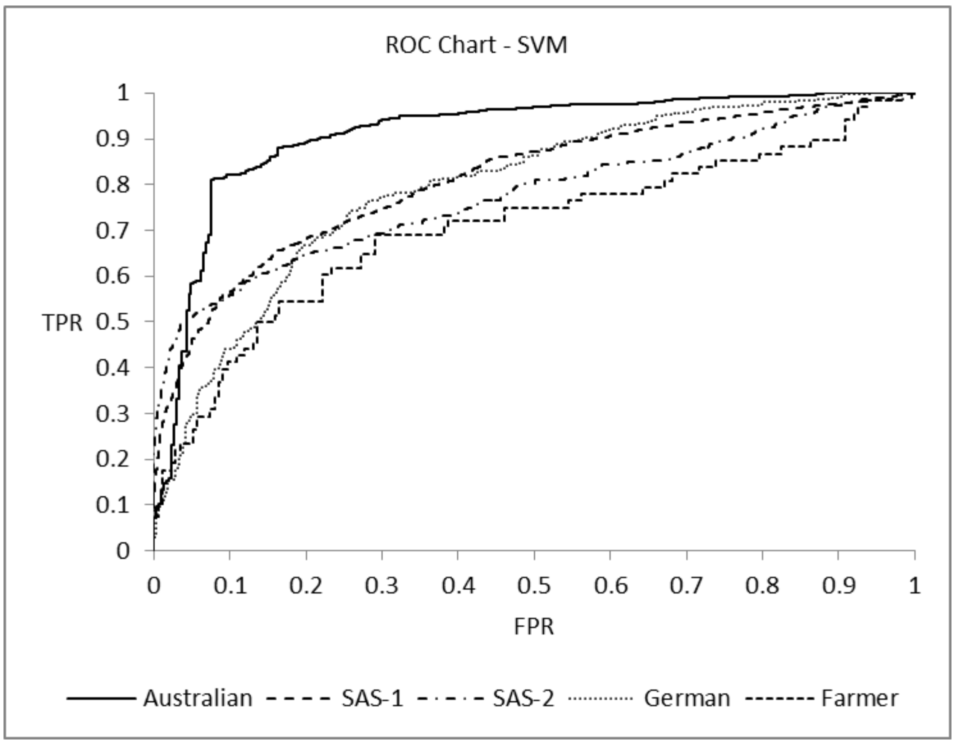


Figure 10. The ROC curves for *k*NN for the 5 datasets.

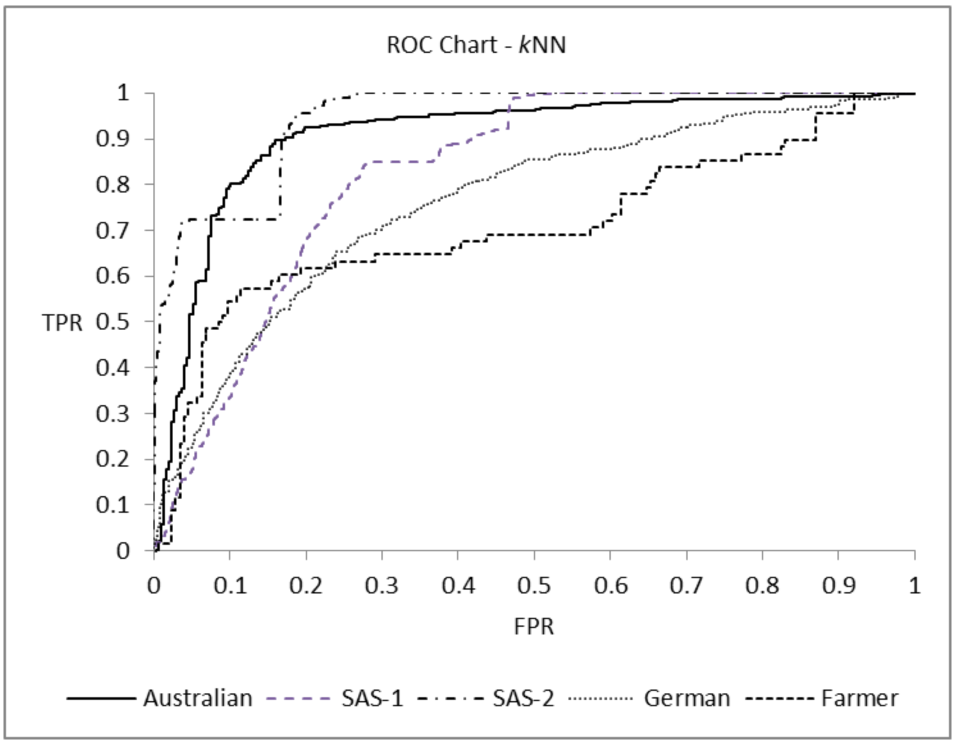
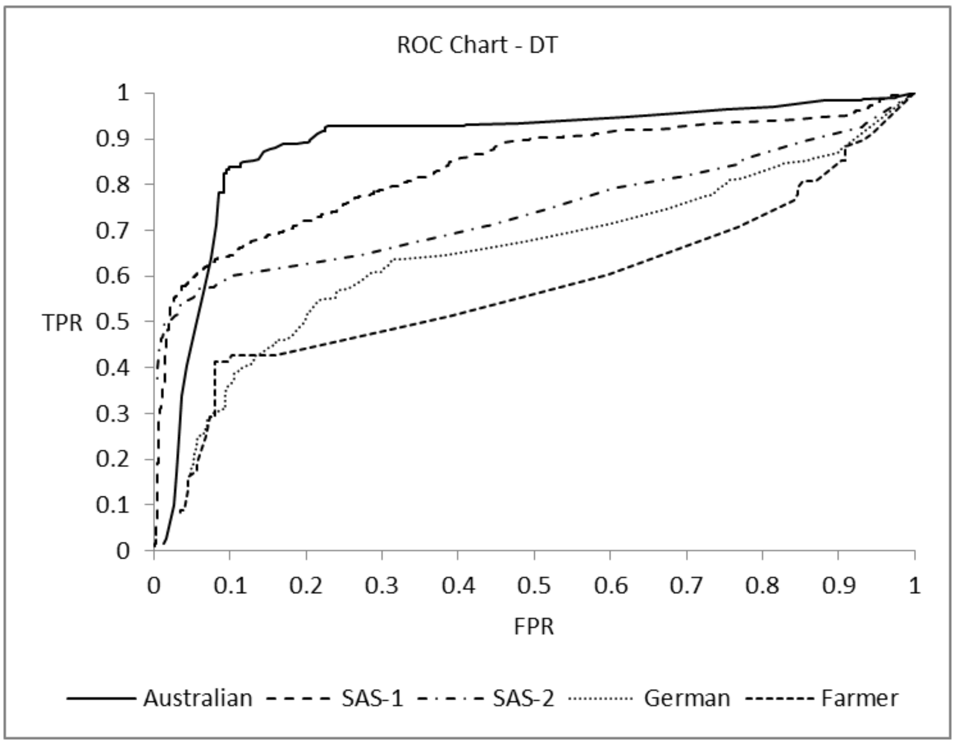


Figure 11. The ROC curves for DT for the 5 datasets



CONCLUSION AND IMPLICATIONS

Accuracy in the detection of loan defaults is crucial to the financial health of loan granting institutions. While the building of reliable credit scoring models has received a great deal of attention from researchers and practitioners in the last few decades, the recent turmoil in the credit lending industry and the consequences on the broader economy have seen credit lending institutions becoming extremely risk averse and reluctant to extend credit, therefore making such modeling even more relevant. In this study we assessed the performances of six known models on five real world datasets that were obtained from different financial settings. We also assessed the quality of the datasets on which the models were constructed. Specifically, in the analysis we first examined the models' classification abilities at a standard 0.5 operating cutoff point with respect to the overall correct classification accuracy rates of bad loans and good loans. We also considered the areas under the ROC charts because they show the overall discriminating ability of the models. In addition we examined the charts themselves as the they can shed some insight into the specific performance of the models at lower or higher cutoff points, a quality that has more utility in practice and thus likely to be used by financial institutions because loan granting institutions do not necessarily use models which perform best at a cutoff point equal to 0.5.

There are several important implications from our study. We found that there are differences between the global performances of the models on each individual dataset. For example, NN and DT do very well when built on the SAS-1 dataset, whereas *k*NN does well for the SAS-2 dataset (Table 7). The SAS datasets are three to five times larger than the next largest dataset in our experiments, the German dataset, and heavily predominated by the good loans. SVM performs the best on the Australian and German datasets. These datasets are medium sized and relatively more balanced. If one looks at the areas under the curves for all the six models, averaged over the five datasets, there are only small differences in the performances between the models. *k*NN (83.2%), NN (81.8%), and SVM (81.0%) slightly outperform LR (80.4) and RBFNN (80.2), but DT (74.6%) lags significantly. However, we recognize that even slight improvements in accuracy of predicting creditworthiness can generate substantial revenues or losses for financial institutions. The poor overall performance of DTs, as per ROC curves, is interesting as these are the models that are easiest to interpret. Moreover, if a financial institution is obliged by law to provide a clear explanation to borrower applicants why a loan is denied, as is required in provisions of the Equal Credit Opportunity Act (ECOA, 1975), the DTs' interpretability and readily understood if-then rules may pose a choice dilemma for practitioners. At the same time, DTs turned out to be good at detecting bad loans at higher operating points. Thus DTs may be a suitable model when a lending institution has high collateral requirements and applies a high or generous cutoff point, even though in this simulation they performed poorly on average when measured with the more global/overall metric of areas under ROC curves. It is also important to note that, for datasets predominated by good loans, DT's performance at the standard 0.5 cutoff point in bad loan classification was better in some cases than those of the other models and it was not significantly worse in the rest of the cases. This is important because many of the datasets in this domain will be gathered from healthy financial institutions where it is more likely that the dataset will contain mostly good loans.

With respect to the data quality, we found that the Australian dataset, which has been used in only a dozen of studies, has the best quality and most ideal characteristics in general. The models constructed on this dataset *consistently* exhibit the highest classification accuracy rates with the average area under the ROC curves equal to 90.9%, the only (higher) exception is the *k*NN model that is built on the SAS-2 dataset (94.2%). In general, models built on SAS-1, SAS-2, and the German datasets perform gradually worse: 82.3%, 80.9%, and 76.0%, respectively, with a few exceptions: The NN and DT models built on the unbalanced SAS-1 dataset stand out, whereas the models built on the Farmer dataset provided by FSA containing financial attributes of the farmers appear to be the worst (70.9%). If one, however, looks at the overall correct classification accuracy rates at a standard 0.5 cutoff point, the SAS-2 dataset (92.8%) appears to have the most favorable qualities. This is not surprising as it is due to excellent classification rates for good loans. The German dataset (74.9%) somewhat surprisingly appears to be the worst (Table 4), worse than the smaller and less balanced Farmer (77.5%) dataset using the areas under the ROC curves. It may be due to the fact that the German dataset, though balanced and well-sized, contains many categorical variables taking distinct levels, and each level is represented by a dummy variable in the models. On the other hand, when the goal is to detect bad loans at the standard 0.5 cutoff point, it is clear that a balanced dataset is important to performance, because when bad loans are underrepresented all these models perform rather poorly.

Data quality: when looked at from a technical, algorithmic performance point of view, we can conclude that the German dataset is a poor quality dataset, that is, the data attributes aren’t good predictors of the classes (i.e. the state of default), in spite of the fact that the dataset is richer than competing datasets in the study and therefore that, contextually, may in fact capture more important socio-economic and/or demographic data that are not necessarily good descriptors of credit default but nevertheless important in practice. However, the models built on the German dataset have too many input variables, including dummy variables, and this may be the reason for the poor performance.

From our results, it is also evident that large size of the dataset is not alone an unqualified positive characteristic. After all, the best dataset, the Australian, is actually the fourth largest dataset out of five. The German dataset, a large dataset (Fayyad & Irani, 1992) performs very poorly. The types of attributes also do not seem to have a definitive impact in the quality of the dataset as both the German and Australian datasets have an equal mix of nominal and numeric attributes and yet the German dataset is much poorer; and the SAS data and the Farmer datasets both contain predominantly numeric data and yet perform very differently. The Farmer dataset has almost exclusively continuous variables; it would however appear that a dataset with exclusively financial ratios as its attributes is not ideal for credit default classifications, as qualitative information about loan applicants is also needed. On the other hand, the ratio between good loans and bad loans seems a good predictor of data quality. When datasets are predominated by good loans, as they likely will be in reality, the more susceptible they are at describing credit defaults poorly.

Analysis of variable importance or worth sheds more light into the relevance of variables in the credit scoring domain. Our study shows that in general, financial attributes of customers are more important than personal, social and employment ones for the prediction task. However this does not suggest practitioners should go out and collect exclusively financial data. We note that the exclusive use of continuous variable data is not well-suited to DTs, which we found are generally better at predicting bad loans.

The models used in this study are well-known and have been used widely and in many contexts and application areas including credit scoring. To the best of our knowledge, however, no credit scoring study has undertaken an in-depth comparative examination of these models within the context of different data settings. The contribution of our study is that it offers a more nuanced and contextualized understanding of the application of these models within different data settings at the standard 0.5 operating cutoff point as well as overall global metrics. This is a contribution, because our analysis yields results that are prescriptively more useful for the practitioner. For example, a finding that NN are better classifiers of bad loans (Chen & Huang, 2003) is incomplete and not practically useful where such a finding is grounded on an “ideal” data set. We believe practitioners are better served by model performance prescriptions that show that model performance is contingent on the nature of the dataset because the ideal dataset, a well-balanced dataset, is improbable in reality. For example NNs are better classifiers of bad loans on well-balanced datasets, at the 0.5 cutoff point. However, DTs are better classifiers of bad loans on unbalanced data sets with or without missing values. See Table 9 for a summary of the major findings of this study.

Table 9. The summary of the major findings from this study.

Data Set	0.5 Cutoff Better Models	Lower cutoffs Better models	Higher cutoffs Better models	Bad Loan avg. classification (Better models)
Australian (medium sized, balanced)	SVM	Model differences indistinguishable	Model differences indistinguishable	RBFNN, DT
SAS-1 (largest, unbalanced, missing values)	NN, DT, *k*NN	*k*NN	NN, DT	NN, DT
SAS-2 (larger, unbalanced, no missing values)	*k*NN and RBFNN	*k*NN	DT, SVM, *k*NN	DT
German (large, more balanced, more attributes)	SVM	SVM	SVM	NN, SVM
Farmer (smallest, unbalanced, real values only)	NN, SVM, *k*NN comparable to LR	*k*NN	Model differences indistinguishable	NN, SVM

In reality, the contingencies are multiple. This paper only begins to scratch the surface. Future studies can explore additional data contingencies. Our findings also suggest a need for more evidence for how these models perform at cutoff points other than the custom 0.5. Future research could examine performance at 0.1, 0.2, 0.3 cutoff points etc. Such evidence would better serve practitioners who may desire to use different measures to assess the attendant risks within their given data contexts. Future research can also extract if-then rules from ensuing models to improve their direct utility in the loan granting decision process.

REFERENCES

Aha, D., & Kibler, D. (1991). Instance-based Learning Algorithms. *Machine Learning, 6*, 37-66.

Baesens, B., Setiono, R., Mues, C., & Vanthienen, J. (2003). Using neural network rule extraction and decision tables for credit-risk evaluation. *Management Science, 49*(3), 312-329.

Baesens, B., Van Gestel, T., Stepanova, M., Van den Poel, D., & Vanthienen, J. (2005). Neural network survival analysis for personal loan data. *Journal of the Operational Research Society, 56*(9), 1089-1098. doi: 10.1057/palgrave.jors.2601990

Barney, D. K., Graves, O.F. , & Johnson, J.D. (1999). The Farmers Home Administration and Farm Debt Failure Prediction. *Journal of Accounting and Public Policy, 18*(2), 99-139.

Belloti, T., & Crook, J. (2009). Support Vector Machines for Credit Scoring and Discovery of Significant Features. *Expert Systems with Applications, 35*(2), 3302-3308.

Ben-David, A., & Frank, E. (2009). Accuracy of machine learning models versus "hand crafted" expert systems - A credit scoring case study. *Expert Systems with Applications, 36*(3), 5264-5271. doi: 10.1016/j.eswa.2008.06.071

Bikakis, N, Gioldasis, N, Tsinaraki, C, & Christodoulakis, S. (2009). Semantic Based Access over XML Data. *Proc. of 2nd World Summit on Knowledge Society*.

Boros, E., Hammer, P.L., Ibaraki, T., Kogan, A., Mayoraz, E., & Muchnik, I.B. (2000). An Implementation of Logical Analysis of Data. *IEEE Transactions on Knowledge Data Engineering, 12*(2), 292-306.

Chen, M. C., & Huang, S. H. (2003). Credit scoring and rejected instances reassigning through evolutionary computation techniques. *Expert Systems with Applications, 24*(4), 433-441. doi: 10.1016/s0957-4174(02)00191-4.

Chen, W. M., Ma, C. Q., & Ma, L. (2009). Mining the customer credit using hybrid support vector machine technique. *Expert Systems with Applications, 36*(4), 7611-7616. doi: 10.1016/j.eswa.2008.09.054

Chrzanowska, M., Alfaro, E., & Witkowska, D. (2009). The individual borrowers recognition: Single and ensemble trees. *Expert Systems with Applications, 36*(3), 6409-6414. doi: 10.1016/j.eswa.2008.07.048

Eggermont, J., Kok, J.N., & Kosters, W.N. (2004). Genetic Programming for Data Classification: Partitioning the Search Space. Paper presented at the *SAC '04 Proceedings of the 2004 ACM Symposium on Applied Computing*.

Espejo, Pedro G., Ventura, Sebastian, & Herrera, Francisco. (2010). A Survey on the Application of Genetic Programming to Classification. *IEEE Transactions on Systems, Man, and Cybernetics—Part C: Applications and Reviews, 40*(2).

Fayyad, Usama M. , & Irani, Keki B. . (1992). On the Handling in Decision Tree of Continuous-Valued Attributes Generation. *Machine Learning, 8*, 87-102.

Finlay, S. (2009). Are we modelling the right thing? The impact of incorrect problem specification in credit scoring. *Expert Systems with Applications, 36*(5), 9065-9071. doi: 10.1016/j.eswa.2008.12.016

Foster, B.P., Zurada, J., & Barney, D. (2010). Could Decision Trees Help Improve Farm Service Agency Lending Decisions? *Academy of Information and Management Sciences Journal., 13*(1), 69-91.

Hall, M. A. (1998). *Correlation-based Feature Subset Selection for Machine Learning.* University of Waikato, Hamilton, New Zealand. .

Henley, W. E., & Hand, D. J. (1996). A k-nearest neighbour classifier for assessing consumer credit risk. *The Statistician, 45*(1), 77-95.

Hevner, Alan R., March, Salvatore T., Park, Jinsoo, & Ram, Sudha. (2004). Design Science in Information Systems. *MIS Quarterly, 28*(1), 75-105.

Huang, C. L., Chen, M. C., & Wang, C. J. (2007). Credit scoring with a data mining approach based on support vector machines. *Expert Systems with Applications, 33*(4), 847-856. doi: 10.1016/j.eswa.2006.07.007

Huang, J. J., Tzeng, G. H., & Ong, C. S. (2006). Two-stage genetic programming (2SGP) for the credit scoring model. *Applied Mathematics and Computation, 174*(2), 1039-1053. doi: 10.1016/j.amc.2005.05.027

Kecman, V. . (2001). *Learning and Soft Computing: Support Vector Machines, Neural Networks, and Fuzzy Logic Models*. Boston, MA: Massachusetts Institute of Technology.

Keerthi, S.S., Shevade, S.K., Bhattacharyya, C., & Murthy, K.R.K. (2001). Improvements to Platt's SMO Algorithm for SVM Classifier Design. *Neural Computation, 13*(3), 637-649.

Khashman, A. (2009). A Neural Network Model for Credit Risk Evaluation. *International Journal of Neural Systems, 19*(4), 285-294.

Krishnamurthi, M., (2007). Improving Credit Card Operations with Data Mining Techniques. *Journal of International Technology and Information Management*, 16(4), 43-60.

Kumar, L, Pandey, A., Srivastava, S., & Darbari, M. (2011). A Hybrid Machine Learning System for Stock Market Forecasting. *Journal of International Technology and Information Management*, 20(1), 39-48.

Laha, A. (2007). Building contextual classifiers by integrating fuzzy rule based classification technique and k-NN method for credit scoring. *Advanced Engineering Informatics, 21*(3), 281-291. doi: 10.1109/tcad.2006.12.004

le Cessie, S., & van Houwelingen, J. C. . (1992). Ridge Estimators in Logistic Regression. *Applied Statistics, 41*(1), 191-201.

Lee, T. S., & Chen, I. F. (2005). A two-stage hybrid credit scoring model using artificial neural networks and multivariate adaptive regression splines. *Expert Systems with Applications, 28*(4), 743-752. doi: 10.1016/j.eswa.2004.12.031

Lee, Tian-Shyug, Chiu, Chih-Chou, Chou, Yu-Chao, & Lu, Chi-Jie. (2006). Mining the customer credit using classification and regression tree and multivariate adaptive regression splines. *Computational Statistics & Data Analysis, 50*(4), 1113-1130.

Lenat, D. B. and Guha, R. V. *Building Large Knowledge Bases*. Addison-Wesley, Reading, Mass., 1990.

Li, S. T., Shiue, W., & Huang, M. H. (2006). The evaluation of consumer loans using support vector machines. *Expert Systems with Applications, 30*(4), 772-782. doi: 10.1016/j.eswa.2005.07.041

Lim, Michael Mah-Hui. (2008). Old wine in new bottles: subprime mortgage crisis – causes and consequences. *The Journal of Applied Research in Accounting and Finance, 3*(1), 3-14.

Luo, S. T., Cheng, B. W., & Hsieh, C. H. (2009). Prediction model building with clustering-launched classification and support vector machines in credit scoring. *Expert Systems with Applications, 36*(4), 7562-7566. doi: 10.1016/j.eswa.2008.09.028

March, Salvatore T., & Smith, G.F. (1995). Design and Natural Science Research on Information Technology. *Decision Support Systems, 15*(4), 251-266.

Mitchell, T. M. (1997). *Machine Learning.* Boston, MA: WCB/McGraw-Hill.

Ong, C. S., Huang, J. J., & Tzeng, G. H. (2005). Building credit scoring models using genetic programming. *Expert Systems with Applications, 29*(1), 41-47.

Owen, A. (2003). Data squashing by empirical likelihood. *Data Mining and Knowledge Discovery, 7*(1), 101-113.

Paleologo, G., Elisseeff, A., & Antonini, G. (2010). Subagging for credit scoring models. *European Journal of Operational Research, 201*(2), 490-499. doi: 10.1016/j.ejor.2009.03.008

Platt, J. (1998). Machines Using Sequential Minimal Optimization. In B. Schölkopf, C. Burges & A. Smola (Eds.), *Advances in Kernel Methods: Support Vector Learning*: MIT Press.

Quinlan, J. R. (1987). Simplifying Decision Trees. *International Journal of Man-Machine Studies, 27*(3), 221-234.

Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*. San Mateo, California.: Morgan Kaufmann.

Sohn, S. Y., & Kim, H. S. (2007). Random effects logistic regression model for default prediction of technology credit guarantee fund. *European Journal of Operational Research, 183*(1), 472-478. doi: 10.1016/j.ejor.2006.10.006

Standifird, S. S., & Marshall, R. S. (2000). The transaction cost advantage of guanxi-based business practices. *Journal of World Business, 35*(1), 21-42.

Stepanova, M., & Thomas, L. (2002). Survival analysis methods for personal loan data. *Operations Research, 50*(2), 277-289.

Thomas, L.C. (2002). A survey of credit and behavioral scoring: Forecasting financial risk of lending to consumers. *International Journal of Forecasting, 16*, 149-172.

Tsai, C. F. (2008). Financial decision support using neural networks and support vector machines. *Expert Systems, 25*(4), 380-393. doi: 10.1111/j.1468-0394.2008.00449.x

Urdaneta, Guido, Colmenares, Juan A., Queipo, Nestor V., Arape, Nelson, Arevalo, Carlos, Ruz, Mirche, . . . Romero, Andreina. A reference software architecture for the development of industrial automation high-level applications in the petroleum industry. *Computers in Industry, In Press, Corrected Proof.*

Vapnik, V. N. (1998). *Statistical Learning Theory*. New York: Wiley.

West, D. (2000). Neural network credit scoring models. *Computers & Operations Research, 27*(11-12), 1131-1152.

West, D., Dellana, S., & Qian, J. X. (2005). Neural network ensemble strategies for financial decision applications. *Computers & Operations Research, 32*(10), 2543-2559. doi: 10.1016/j.cor.2004.03.017

Witten, I.H., & Frank, E. (2005). *Data Mining: Practical Learning Tools and Techniques* Morgan Kaufmann Publishers.

Yu, L., Wang, S. Y., & Lai, K. K. (2009). An intelligent-agent-based fuzzy group decision making model for financial multicriteria decision support: The case of credit scoring. *European Journal of Operational Research, 195*, 942-959. doi: 10.1016/j.ejor.2007.11.025

Yuan, Ruixi, Li, Zhu, Guan, Xiaohong, & Xu, Li. (2010). An SVM-based machine learning method for accurate internet traffic classification. *Information Systems Frontier*, (12), 149–156.

Zhou, L. G., Lai, K. K., & Yu, L. A. (2008). Credit scoring using support vector machines with direct search for parameters selection. *Soft Computing - A Fusion of Foundations, Methodologies and Applications*, *13*(2), 149-155. doi: 10.1007/s00500-008-0305-0

Zurada, J. (2007). Rule Induction Methods for Credit Scoring. *Review of Business Information Systems, 11*(2), 11-22.

Zurada, J. (2010, January 5-8). Could Decision Trees Improve the Classification Accuracy and Interpretability of Loan Granting Decisions? Paper presented at the *43rd Hawaii International Conference on System Sciences (HICSS'2010)*, Hawaii.

Zurada, J., & Kunene, N. (2010). Performance Assessment of Data Mining Methods for Loan Granting Decisions: A Preliminary Study. Paper presented at the *Artificial Intelligence and Soft Computing -10th International Conference on Artificial Intelligence and Soft Computing (ICAISC 2010).*

Zurada, J., & Kunene, N. (2011, January 4-7). Comparisons of the Performance of Computational Intelligence Methods for Loan Granting Decisions. Paper presented at the *44th Hawaii International Conference on System Sciences (HICSS'44)*, Kauai, HI.