Intelligent Data Analysis 17 (2013) 1115–1134
DOI 10.3233/IDA-130623
IOS Press

Impact of noise on credit risk prediction: Does data quality really matter?

Bhekisipho Twala
Department of Electrical and Electronic Engineering Science, University of Johannesburg, P O Box 524, Auckland Park, Johannesburg 2006, South Africa
E-mail: btwala@uj.ac.za

Abstract. Machine learning has been successfully used for credit-evaluation decisions. Most research on machine learning assumes that the attributes of training and tests instances are not only completely specified but are also free from noise. Real world data, however, often suffer from corruptions or noise but not always known. This is the heart of information-based credit risk models. However, blindly applying such machine learning techniques to noisy financial credit risk evaluation data may fail to make very good or perfect predictions. Unfortunately, despite extensive research over the last decades, the impact of poor quality of data (especially noise) on the accuracy of credit risk has attracted less attention, even though it remains a significant problem for many. This paper investigates the robustness of five machine learning (supervised) algorithms to noisy credit risk environment. In particular, we show that when noise is added to four real-world credit risk domains, a significant and disproportionate number of total errors are contributed by class noise compared to attribute noise; thus, in the presence of noise, it is noise on the class variable that are responsible for the poor predictive accuracy of the learning concept.

Keywords: Noisy data, credit risk, machine learning, classifiers, predictive accuracy

1. Introduction

The implementation of Basel II [11] has not only produced an unprecedented amount of data on banks' credit portfolios but increased banks' data quality requirements. Ensuring the quality of this data and managing data ownership appropriately is a fundamental prerequisite for sound credit risk management. Data quality refers to correct, complete, and timely information that is available for a specific analytical use. It is not an end in itself nor can it be measured on an absolute scale; rather, it is subject to continuous change and new challenges. Data deficiencies increase transaction costs and underscore the importance of data quality as a factor in achieving competitive advantage. Whether triggered by regulatory requirements or by competitive market pressure, data quality is an asset that influences an institute's ability to reach its targets. Consequently, measures that aim at improving data quality represent investments that can be justified by their contribution to the achievement of performance goals.

Over the last decade, a number of the world's largest banks have developed sophisticated systems (including supervised learning) in an attempt to model the credit risk arising from important aspects of their business lines. Such models are intended to aid banks in quantifying, aggregating and managing risk across geographical and product lines. The outputs of these models also play increasingly important roles in banks' risk management and performance measurement processes. Thus, credit risk modelling may indeed prove to result in better internal risk management and may have the potential to be used in supervisory oversight of banking organisations.

One of the most important feature of a problem domain as the application of supervised learning (SL) is concerned, is the form that the data takes. Most learning techniques [3,14,15,20,25,51] that have actually been applied assume that the data are presented in a simple attribute-value format in which a record has a fixed number of constant-values fields or properties. Another important feature of a problem domain is the quality of data available. Most "real" data is imperfect [37,57]: incomplete (missing values for some attributes and instances), irrelevant (some fields that do not relate to the problem at hand), redundant (involving unknown, or at least unexpressed, relations between the attributes), noisy (for example, some attributes have inherent measurement errors) and occasionally erroneous (e.g., incorrectly transcribed). This paper only focuses on noisy data.

It has generally been recognized that data quality is a point of major concern for successfully implementing credit risk management (modelling) systems. Noise is one important aspect of data quality that my impact interpretations of the data, models created from the data and decision based on the data. Noise can reduce system performance in terms of predictive accuracy, time in building a supervised learning algorithm (which from now on we shall be used interchangeable with classifier) and the size of the classifier [5,22,31,35,64]. While these studies have focussed on issues pertaining data quality, there has only been limited work done on the impact of noise in credit risk prediction. This could be attributes to the fact that researchers fail to realize that data quality is an important component of SL tools. Instead more focus is spent on the learning algorithms and methods themselves. In fact, most research in these fields begins with the assumption that the data feeding the algorithms is of high quality (i.e. accurate, compete and timely). However, other fields such as marketing, economics, education, statistics, psychometrics and medicine or biometrics have begun to investigate the issue.

There are many kinds of "noise" that could occur in the instances. These include errors, spurious correlations (i.e. correlations that are due mostly to the influences of one or more "other" variables), attributes that are not recorded, two instances having the same attribute/value pairs but different classifications, some values of attributes being incorrect because of errors in the data acquisition process or the processing phase, values of attributes being missing, and the classification or class label being wrong (for example, 1 instead of 2) because of some error. Monago and Kodratoff [44] present a more detailed analysis of the sources of noise in data.

Noise is often similarly divided into two major categories that are classification noise (misclassifications/mislabelling or training and test instances with incorrect class labels) feature noise (errors introduced to attribute values). Zhu and Wu [69] propose to distinguish the following examples of attribute errors: erroneous attribute values, missing so-called 'do not know' values, and incomplete or so-called 'do not care' values. The two major types of classification noise are contradictory instances (instances with the same values of the attributes but different class labels, forming so-called irreducible or Bayes error) and wrongly classified (labelled) instances that are misclassifications (mislabellings). The errors may naturally occur in credit scoring when different classes have very similar or even overlapping credit history.

There are three key aspects in this scenario. First, if all attribute values were known then the learner could make perfect predictions after learning the target concept. Second, in any given example provided to the learner, some of the attributes are likely to be unspecified. Finally, the learner's goal is to make correct prediction if the values for the provided attributes are sufficient to do so, or otherwise indicate that insufficient information has been provided.

In the practice of SL, learning data typically contain errors. Imperfections in data can be due to various, often unavoidable causes. This include: hardware failure, programming errors, measurement errors, human mistakes, and perception errors by domain expert. We refer to this source of noise as misspecification. The second type of noise (which is a fairly common source of noise in real world applications)

is called residual variation by Mingus. Residual variation refers to additional factors that affect results, but are not recorded mainly because those recording the data were unaware of the effect or they simply unable not record them. The result with either source of noise are instances that are either inconsistent or that have features that should correlate with the category but do not because of noise masking correlation. Noise can also come from the treatment of missing values, when an example with unknown attribute value is replaced by a set of weighted examples corresponding to the probability distribution of the missing value. The typical consequences of noise in learning data are: (a) low prediction accuracy of induced hypotheses on new data, and (b) large hypotheses that are hard to interpret and to understand by the user. In addition, one of the major problems of SL algorithms that do not account for noise is that they may tend to overfit the data.

There are three main approaches of the handling noise. First, the imperfections can be ignored and left in the source unchanged (robust learning), second the imperfections can be removed, (filtered) or third the imperfections can be corrected (correction). Ignoring the imperfections and leaving the noise in the data set would require a robust algorithm [29] Excluding the imperfections from the dataset limits the impact of spurious findings but is information inefficient because less data is available for the analysis. This approach works by filtering the data according to criteria and removing the cases which do not adhere with the criteria and was successfully used by Brodley and Friedl [8,9]. Polishing is one correction approach followed by Teng, which attempts to preserve the maximal information available (by identifying and repairing the misclassified or noisy instances) when building the classifier.

There are other credit score techniques like FICO score [21] from the Fair Isaac Corporation that performs well with noisy data. The FICO score is a highly predictive model designed to evaluate consumer credit risk on any established for of credit bureau data. Each score is based on information the credit bureau keeps on file about an individual. As this information changes, your credit scores tend to change as well. The FICO score affect both how much and what loan terms (interest rate, etc.) lenders will offer an applicant at any given time. Taking steps to improve your FICO scores can help one qualify for better rates from lenders.

It is also interesting to know how the accuracy changes relative to the classifier, as test sets contain more and more noise on the attributes. [47] demonstrated the impact of noise on predictive accuracy. He showed how the removal of noise from attribute information decreases predictive accuracy of the resulting classifier, especially if the same attribute noise is present in the data to be classified. Brodley and Friedl [8,9] illustrated that for classification noise levels of less than 40%, removing mislabelled instances from the training data resulted in higher predictive accuracy relative to classification accuracies achieved without "cleaning" the training data. In some real world applications, unseen (test) instances contain a lot of noise, while the training set contains relatively noise free information. For example, in credit scoring, the training data from historical records of credit applicants could contain almost noise free attributes. However, before classifying a new applicant, it is always desirable to perform a preliminary check before the results of some time-consuming credit check becomes available if time is crucial, or before conducting some very expensive credit check if cost is important. In this situation, classification needs to be performed on unseen instances (new applicants) with noisy attributes.

To this end, the major contribution and uniqueness of the work presented in this paper is to:

- Show the robustness (sensitivity and reduction capacity) of five of the well-known classifiers to noise in terms of credit risk predictive accuracy; different proportions of noise are considered for this task.
- Further demonstrate the impact of different types of noise (attribute or class) on credit risk predictive accuracy in either the training or test set or in both training and test sets.

The remainder of the paper is organized as follows. The next section discusses five different classifiers that are used as the basis of our study. This is followed by a description of the technique used to emulate noise on various attributes and at different levels is described. Then we describe our analysis, followed by the empirical comparison of the five classifiers tested against artificially-noise-polluted datasets, for different types of noise (attribute and class) and at different levels of noise. The paper concludes with discussion of the significance of our results and how the research could be progressed.

2. Classifiers

The most important feature of a problem domain, as far as the application of classifiers are concerned, is the form that the data takes and the quality of the data available. Our main focus will be on the latter. The problem of handling noise has been the focus of much attention in the banking and finance communities. Specific classifiers that are known to be robust enough to cope with noisy data, and to discover laws in it that may not always hold but are useful for the problem at hand, are now going to be described. The five classifiers (artificial neural network, decision tress, naïve Bayes classifier, k-nearest neighbour and logistic regression) which are used in all experiments in the paper are now described.

2.1. Artificial neural networks

Artificial Neural Networks (ANNs), usually nonparametric approaches, are represented by connections between a very large number of simple computing processors or elements (neurons), have been used for a variety of classification and regression problems. These include pattern and speech recognition [52,54,67], credit risk prediction [4,46,55,62,66], and so on. There are many types of ANNs, but for the purposes of this study we shall concentrate on single unit perceptrons and multi-layer perceptrons also known as "backpropagation networks". The backpropagation learning algorithm performs a hill-climbing search procedure on the weight space described above or a (noisy or stochastic) gradient descent numerical method whereby an error function is minimised. At each iteration, each weight is adjusted proportionally to its effect on the error. One cycle through the training set and on each example changes each weight proportionally to its effect on lowering the error. One may compute the error gradient using the chain rule and the information propagates backwards through the network through the interconnections, which accounts for the procedure's name.

2.2. Decision trees

A decision tree (DT) [7,47,50,63] is a model of the data that encodes the distribution of the class label in terms of the predictor attributes; it is a directed, acyclic graph in a form of a tree. The root of the tree does not have any incoming edges. Every other node has exactly one incoming edge and zero or more outgoing edges. If a node n has no outgoing edges we call n a leaf node, otherwise we call n an internal node. Each leaf node is labelled with one class label; each internal node is labelled with one predictor attribute called the splitting attribute. Each edge e originating from an internal node n has a predicate q associated with it where q involves only the splitting attribute of n. There are two ways to control the size of the tree. For a bottom-up pruning strategy, a very deep tree is constructed, and this tree is cut back to avoid overfitting the training data. For top down pruning, a stopping criterion is calculated during tree growth to inhibit further construction of parts of the tree when appropriate. In this paper we follow the bottom up strategy. A DT can be converted into rules which could be used for prediction tasks such as credit default and bankruptcy [41].

2.3. Naïve Bayes

The naïve Bayes classifier (NBC) is perhaps the simplest and most widely studied probabilistic learning method. Such algorithm has been applied most recently for credit risk modelling [28]. It learns from the training data, the conditional probability of each attribute A_i, given the class label C [19,43,52]. The strong major assumption is that all attributes A_i are independent given the value of the class C. Classification is therefore done applying Bayes rule to compute the probability of C given $A_1, \ldots, A_n$ and then predicting the class with the highest posterior probability. The independence assumption is fairly strong and is often not applicable. When the strong attribute independence assumption is violated, the performance of the NBC might be poor. However, Domingos and Pazzani [18] argue that the NBC is still optimal even when the independence assumption is violated as long as the ranks for the conditional probabilities of classes given and instance are correct. A few techniques [33,36] have been developed to improve the performance of the NBC.

2.4. k-Nearest neighbour [19,23]

One of the most venerable algorithms in statistical pattern recognition is the nearest neighbour (NN). Of late, such an algorithm has become popular in credit scoring [24,25,28]. NN methods are sometimes referred to as memory-based reasoning or instance-based learning (IBL) or case-based learning (CBL) techniques and have been used for classification tasks. They essentially work by assigning to an unclassified sample point the classification of the nearest of a set of previously classified points. The entire training set is stored in the memory. To classify a new instance, the Euclidean distance (possibly weighted) is computed between the instance and each stored training instance and the new instance is assigned the class of the nearest neighbouring instance. More generally, these k-nearest neighbours (k-NNs) are computed, and the new instance is assigned the class that is most frequent among the k neighbours. IBL's have three defining general characteristics: a similarity function (how close together the two instances are), a "typical instance" selection function (which instances to keep as examples), and a classification function (deciding how a new case relates to the learned cases) [2]. The lack of a formal framework for choosing the size of neighbourhood "k" can be problematic. Holmes and Adams [27] have proposed a probabilistic strategy to overcome these difficulties.

2.5. Logistic discrimination

Logistic discrimination analysis (LgD), due to Cox [12] and Day and Kerridge [16], is related to logistic regression [26,39]. The dependent variable can only take values of 0 and 1, say, given two classes. This technique is partially parametric, as the probability density functions for the classes are not modelled but rather the ratios between them. The *a posteriori* class probabilities are computed by the logistic distribution. Computational details can be found in Hosmer and Lameshow [26] and Menard [40]. A new element is classified as 0 if the estimated probability $\pi_0 \leqslant c$ and as 1 if $\pi_0 > c$, where c is the cut-off point score. Typically, the cut-off point used could be 0.5 [56]. In fact, it has been argued that the slope of the cumulative logistic probability function is steepest in the region where $\pi_i = 0.5$ [45]. For a prediction problem with more than two classes, multinomial logit models are used [26,34,38].

3. Emulating noise

3.1. Basic concepts

In our study, outliers are more specifically defined as instances with attribute values that reside on

the extremes of the general distributions observed for the given attributes in the data set. In contrast, instances may be considered noisy when the values of one or more attributes of an instance are corrupted or incorrect relative to the values of the other attributes. However, an instance without errors may appear noisy when one or more of its attributes does not follow the general distributions observed for the given attributes. These exceptions therefore often appear as noisy instances. Such instances are very difficult to detect without the input of a domain expert.

Two major types of classification noise are contradictory instances (instances with the same values of the attribute but different class labels) and wrongly classified instances that are misclassifications. These errors naturally occur in credit risk management when different classes have very similar or even overlapping credit scores. In this paper we focus on the second type of classification noise because it is the most common and can refer to several sources of classification noise (including data entry errors, subjectivity and the inadequacy of information used to label each instance in a dataset) [10].

Domains in which credit risk experts may disagree are natural ones for subjective labelling errors. In particular, if in some practical classification problem the absolute ground truth is unknown then experts must subjectively provide labels and mislabelled instances naturally appear.

3.2. Methodology

For the noise model to be applied on the various datasets, two assumptions are considered to be true for all datasets [30]:

1. All the attributes of the datasets (including class) are normally distributed.
2. Noise is randomly distributed and independent from the data.

Then for every case (y_i, x_i) in the dataset L the pair (y_i, x_i) of the dependent variable Y and the matrix of independent variables X is substituted by another (y_i', x_i') by a probability of p, where p is the noise level. The new pair is calculated by the following formula:

$$x'_{ij} = \begin{cases} x_{ij} + \sigma_{xj} z_j & v_{ij} \geqslant p, \\ x_{ij} & v_{ij} < p. \end{cases} \qquad y'_{ij} = \begin{cases} y_{ij} + \sigma_y z_j & v_{ij} \geqslant p, \\ y_{ij} & v_{ij} < p. \end{cases}$$

where, σ_{xj} is the standard deviation of x_j; z_j is a normally distributed random variable and is calculated by the inverse function of density-probability of the normal distribution for a value of v_{ij}, having mean zero and a standard deviation equal to unity; $v_{ij} \in (0, 1)$ is a probability variable produced by a random value generator following a uniform distribution. Thus, $z_{ij} = \textit{norminv}\ (v_{ij})$, $j \in [1, \ldots, k]$. In other words, this process involves converting or transforming scores of an attribute to z-scores (standardisation). A distribution in standard form has a mean of 0 and a standard deviation of 1. However, it is important to note that a z-score transformation changes the central location of the distribution and the average variability of the distribution. It does not change the skewness or kurtosis.

Attribute noise: To introduce $p\%$ attribute feature noise to a dataset of D instances, each of which has F attributes (excluding the class label), we randomly select with replacement $\frac{p*D*F}{100}$ instances with replacement and for each of them we change the value of a randomly selected feature. For nominal features, the new value is chosen randomly with equal probability from the set of all possible values. For numeric features, the new value is generated from a Normal distribution defined by the mean and the standard deviation of the given feature, which are estimated from the dataset. Attribute feature noise was introduced to both the training and test sets.

Classification noise: To introduce $p\%$ classification noise to a dataset of D instances, we randomly select $\frac{p*D}{100}$ instances with replacement and change their class labels to one of the other also chosen randomly with equal probability. Then the data were split into 10 subsets for the stratified 10-fold cross-validation. This class-noise-inducing approach is almost similar to the one followed by Dietterich [17]. Note that the stratification was carried out using the new labels. In addition, class noise was only introduced in the training set and not to the test set as the latter has no class labels.

However, until now, the intrinsic relationship between noise of each attribute and the classification accuracy is unclear, and we still have no idea about what types of attributes are sensitive to noise and why they are more sensitive than others. Therefore, we adopt the χ^2 test to analyse the correlations between each attribute and the class label. Essentially, the test is a widely used method for testing independence and/or correlation between two vectors. It is based on the comparison of observed frequencies with the corresponding expected frequencies. The closer the observed frequencies are to the expected frequencies, the greater is the weight of evidence in favour of independence. The is χ^2 defined by:

$$\chi^2 = \sum \frac{(O_{r,c} - E_{r,c})^2}{E_{r,c}}.$$

where $O_{r,c}$ is the observed frequency count at level r of attribute A and level c of attribute B, and $E_{r,c}$ is the expected frequency count at level r of attribute A and level c of attribut B. A χ^2 value of 0 implies the corresponding two vectors are statistically independent with each other; the higher the χ^2 value, the higher the correlation between the corresponding vectors.

To execute the χ^2 test between an attribute (A_i) and the class label (C), we take each of them as a vector, and calculate how many instances contain the corresponding values.

4. Experiments

4.1. Experimental set-up

One of the objectives of this study is to investigate the behaviour of techniques are well-known to be noise tolerant systems. This section describes experiments that were carried out in order to compare the noise-tolerant performance of five different techniques when noise itroduced to the attributes [both in the training and test (unseen) sets]. In other words, we have crupted training and test sets. The effects of different proportions of attributes and classification noise when building the classifier (training) are further examined, experimentally. Finally, the impact of the nature of different levels of attribute noise on the test set with classification accuracy of resulting classifiers is examined.

In order to empirically evaluate the performance of the five well known classifiers at increasing noise levels, an experiment is used on four credit datasets in terms of misclassification error rate. Each dataset defines a different learning problem as described below.

4.1.1. Loan payments

The data was used to classify a set of firms into those that would default and those that wouldn't default on loan payments. Of the 32 examples for training, 16 belong to the default case and the other 16 to the non-default case. All the 16 holdout examples belong to the non-default case. The 18 attributes in this data are: (1) net income/total assets, (2) net income/sales, (3) total debt/total assets, (4) cash flow/total debt, (5) long-term debt/net worth, (6) current assets/current liabilities, (7) quick assets/sales, (8) quick

assets/current liabilities, (9) working capital/sales, (10) cash at year-end/total debt, (11) earnings trend, (12) sales trend, (13) current ratio trend, (14) trend of L.T.D./N.W., (15) trend of W.C./sales, (16) trend of N.I./T.A., (17) trend of N.I./sales and (18) trend of cash flow/T. D. For a detailed description of this data, the reader is referred to [1].

4.1.2. Texas banks

Texas banks that failed during 1985–1987 were the primary source of data. Data from a year and two years prior to their failure were used. Data from 59 failed banks were matched with 59 non-failed banks, which were comparable in terms of asset size, number of branches, age and charter status. Tam and Kiang had also used holdout samples for both the 1 and 2 year prior cases. The 1 year prior case consists of 44 banks, 22 of which belongs to failed and the other 22 to non-failed banks. The 2 year prior case consists of 40 banks, 20 of which belongs to failed and 20 to non-failed banks. The data describes each of these banks in terms of 19 financial ratios. For a detailed overview of the data set, the reader is referred to [62].

4.1.3. Australian credit approval

This credit card applications data set has 690 observations with 15 attributes. Of the attributes, nine are discrete with two to fourteen values, and six continuous attributes. There are 307 positive instances and 383 negative instances in this data set. One or more attribute values are missing from 37 instances. All attribute names and values have been changed to meaningless symbols to protect confidentiality of the data. This dataset was obtained from the UCI repository of ML [6].

4.1.4. German credit

This data set was also obtained from the UCI repository of ML [6]. It contains 1000 observations on 20 attributes. The class attribute describes people as either good (about 700 observations) or bad (about 300 observations) credits. Other attributes include status of existing checking account, credit history, credit purpose, credit amount, savings account/bonds, duration of present employment, instalment rate in percentage of disposable income, marital status and gender, other debtors/guarantors, duration in current residence, property, age, number of existing credits at this bank, job, telephone ownership, whether foreign worker, and number of dependents.

In all our experiments the Weka machine learning library in Java [68] and default parameter values were always used for all the five classifiers unless otherwise stated here. For example, the k-NN classifier was tuned for $k = 5$. To avoid sampling bias, we perform 10-fold cross validation on every classifier and repeated the process 10 times, each time randomizing the order of the data set. Weka provides platform with methods that implements functions to build and evaluate various classifiers.

To perform the experiment each dataset was split randomly into 10 parts (Part I, Part II, Part III, Part IV, Part V, Part VI, Part VII, Part VIII, Part IX, Part X) of equal (or approximately equal) size. 10-fold cross validation is used for the experiment. For each fold, nine of the parts of the instances in each category are placed in the training set, and the remaining one is placed in the corresponding test. The same splits of the data are used for all the classifiers.

Since the distribution of noise among attributes and class attribute are two of the most important dimensions of this study, two suites of data are created corresponding to attribute noise and classification noise. Feature noise is introduced to both the training and testing sets and has three versions: attribute noise to both the training and testing sets ($\mathrm{ATTR}_{\mathrm{TR/TS}}$); attribute noise to only the training set ($\mathrm{ATTR}_{\mathrm{TR}}$); and attribute noise to only the testing set ($\mathrm{ATTR}_{\mathrm{TS}}$). For the class attribute, noise is artificially modelled only to the training set and not to the test set ($\mathrm{CLASS}_{\mathrm{TR}}$).

In order to simulate noise on both attributes and class, the original datasets are run using the model presented in Section 3.2. A brief description of how class and attributes are corrupted with noise is given below.

Both of these procedures have the same percentage of noise as their parameters. These two approaches are also run to get datasets with four levels of proportion of noise p, i.e., 0%, 15%, 30% and 50% missing values.

It is reasoned that the condition with no noise should be used as a baseline and what should be analysed is not the error rate itself but the increase or excess error induced by the combination of conditions under consideration. Therefore, for each combination of classifiers, the number of attributes with noise, noise on the class attribute, proportion of noise, and the error rate for all data present is subtracted from each of the three different levels of noise. This will be the justification for the use of differences in error rates analysed in some of the experimental results.

All statistical tests are conducted using the MINITAB statistical software program [42]. Analyses of variance, using the general linear model (GLM) procedure [32] are used to examine the main effects and their respective interactions. For attribute (feature) noise, this was done using a 3-way repeated measures design (where each effect was tested against its interaction with datasets). The fixed effect factors were the: classifiers; number of attributes with noise; and proportion of noise. For classification noise, the fixed factors are the same with only the number of attribute noise factor being replaced by classification noise.

The four datasets used were used to estimate the smoothed error. In fact, accuracy of the tree, in the form of a smoothed error rate, was predicted using the test data. Results were averaged across ten folds of the cross-validation process before carrying out the statistical analysis. The averaging was done as a reduction in error variance benefit. We are aware that this is a crude measure since it does not reflect the accuracy of predictions for different classes within the data. Classes are not equally likely, and those with few instances are usually predicted badly. However, like all performance measures, the smoothed error rate has its own strengths and weaknesses.

4.2. Experimental results

The results are presented in two parts. The first part of this section compares the performance of five classifiers and the effect of attribute and classification noise on predictive accuracy. Within the first part we present results of the impact of attribute noise when it occurs on both training and testing sets, then the results of the effect of attribute noise when it is introduced on only the test set. Finally, results of the effect classification noise (and no attribute noise) on predictive accuracy. The second part presents the overall results of each method, averaged for all four datasets.

4.2.1. Experimental results I

For readability and space compactness purposes, Tables 5–9 are presented in the Appendix section. The tables present the accuracy of four datasets for five classifiers and on four different combinations of type of attribute noise, respectively. The first row reports the noise rate used to corrupt the data. Also note that for each dataset, the percentage of the entire training set that is corrupted for a noise rate of x will be less than $x\%$ because only some pairs of classes are considered problematic. The actual percentage of corrupted training data is reported in the second row of each table.

In all the tables we show the accuracy for each individual dataset of each of the five classifiers learned from noise on the attributes to both the training and testing sets ($\mathrm{ATT}_{\mathrm{TR/TS}}$); noise on the attributes to the testing set ($\mathrm{ATTR}_{\mathrm{TS}}$); noise on the attributes to the training set ($\mathrm{ATTR}_{\mathrm{TR}}$); and noise on the class

attribute only to the training set (CLASS_{TR}). From these experimental results, we have the following observations:

1. All the five classifiers significantly reduce predictive accuracy at all levels of noise from 5% to 50% for all four datasets. Otherwise, all the classifiers show a very good fit to the noise free problems (i.e., at the 0% level). In fact, at lower levels of noise the classifier compare favourably. Overall, DT achieves the highest accuracy rates, followed by ANN, NBC and kNN, respectively. The worst overall performance is by LgD. This is the case for most of the datasets.
2. Classification noise appears to have more impact on predictive accuracy compared to attribute noise in the training set, especially for the Loan payments and Texas banks datasets. For Australian credit application and German Credit, the opposite exists although this varies by the type of classifier used. For example, for the Australian credit data, attribute noise on the training set has more impact than class noise when using the ANN classifier and at higher levels of noise while for the German credit dataset this occurs when using the LgD classifier. Also, for German credit, the impact of attribute noise in the testing set is more severe than class noise. Overall, most of the classifier are able to deal with attribute noise in only the testing set well.
3. In terms of the robustness for tolerating attribute noise in both the training and testing sets, all the classifiers appear to be tolerant. In fact, for this condition, NBC outperforms all the classifiers at higher levels of noise (Australian credit dataset). In the case of German credit, this is true for the kNN classifier although a good performance by NBC is also observed at some levels of noise.
4. In summary, all the classifiers suffer decreases in predictive accuracy due to noise level increases. In most cases, the decrease in accuracy is linear with respect to the increase of noise level. Also, the behaviour of different classifiers varies from one dataset to another. In particular DT and ANN have higher accuracies for corrupted data their respective accuracies deteriorate less in comparison with NBC or kNN. NBC, kNN and LgD appear to be less tolerant to class noise, especially at higher levels (30–50%)

4.2.2. Experimental results II

4.2.2.1. Overall results

Figure 1–3 summarises the overall excess error rates for five classifiers on credit risk predictive accuracy. The behaviour of these techniques is explored under varying amounts of two types of imperfections in data. The error rates of each technique are averaged over the 4 datasets.

Main effects

The ANOVA of all significant effect and interaction effects are given in Table 5 in the Appendix. All the main effects were found to be significant at the 5% level of significance ($F = 49.462$, $df = 4$ for classifiers; $F = 163.423$, $df = 3$ for attribute and class noise; $F = 173.917$, $df = 4$ noisy data proportions; $p < 0.05$ for each main effect).

Figure 1 plots the overall excess error rates for the classifiers. From the results it follows that DT achieves the highest accuracy rates (at all levels of noise) with an error rate of 30.7%, followed by ANN (32.8%), NBC (35.4%), and kNN (37.3%). The worst performance is by LgD with an error rate of 41.4%. Also, there appears to be significant differences among the five techniques at the 5% level of significance ($p < 0.05$).

From Fig. 2, it appears that for all the classifiers, classification noise has a greater effect on predictive accuracy compared with attribute noise on the training set and/or test set. For example, the difference in error rate between class noise and attribute noise in both the training and test sets is the two conditions is about 3%. Noise has a lesser impact when it occurs on attributes to both the training and testing sets.

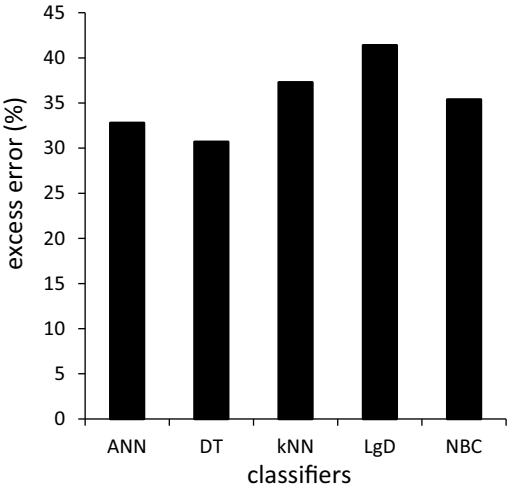

Fig. 1. Overall means for classifiers.

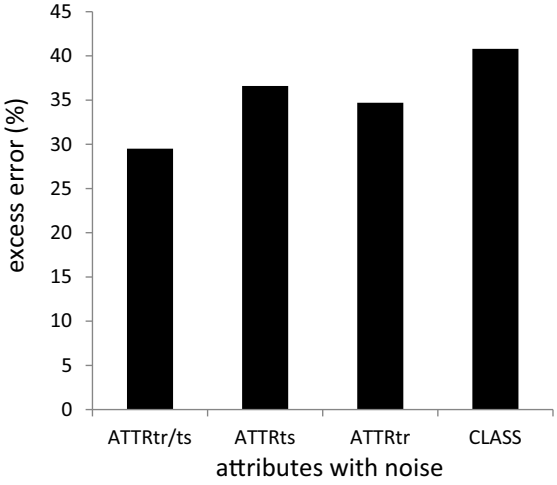

Fig. 2. Overall means for type of attribute noise.

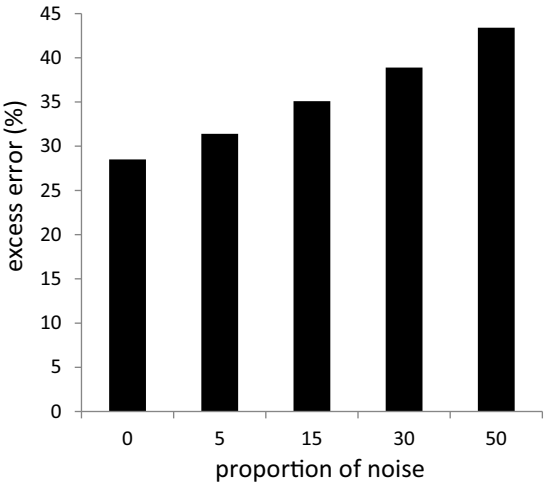

Fig. 3. Overall means for noise proportions.

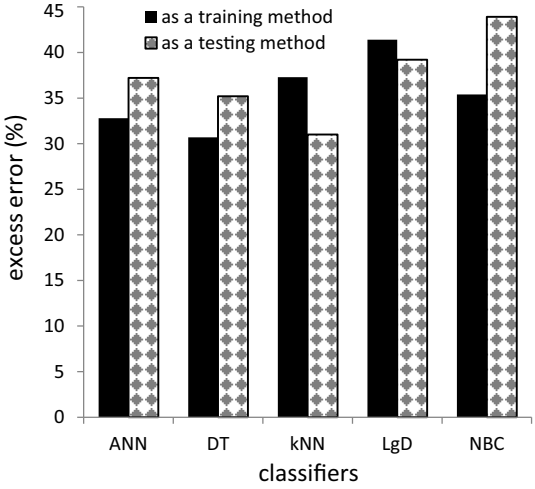

Fig. 4. Overall means for classifiers (training/test).

Increases in noisy data proportions are associated with increases in error rates (Fig. 3). This figure further shows all the techniques degrading in performance as the level of noise increases in attributes and class. Figure 4 shows DT achieving the most superior performance as a training method with an error rate of 30.7%, followed by NBC (32.8%), k-NN (35.4%), LgD (37.3%) and ANN (39.4%), respectively. The best testing method is kNN followed by NBC with error rates of 31.0% and 33.4, respectively. LgD achieves the worst performance for both as a training and testing method (with error rates of 41.4% and 43.9%). All the classifiers appear to deal with noisy training data better compared to when dealing with noisy test data. In other words, the accuracy rates achieved by all the classifiers when dealing with noisy training data are higher compared with the accuracy rates obtained from noisy test data.

Interaction effects

All the two-way interaction effects were found to be significant at the 5% level of significance. Figure 5 presents the results of the interaction between classifiers and the type of attributes and class with noise.

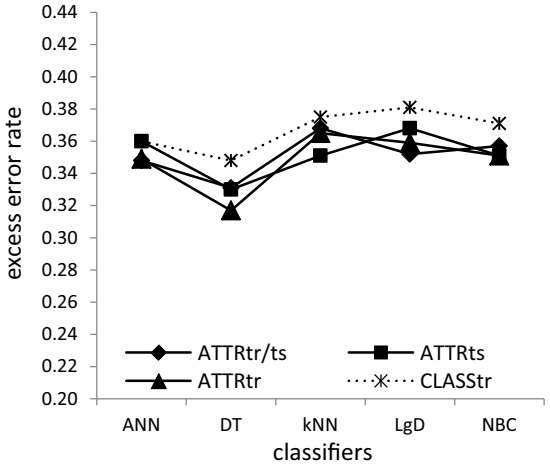

Fig. 5. Interaction between classifiers and type of attribute and class with noise.

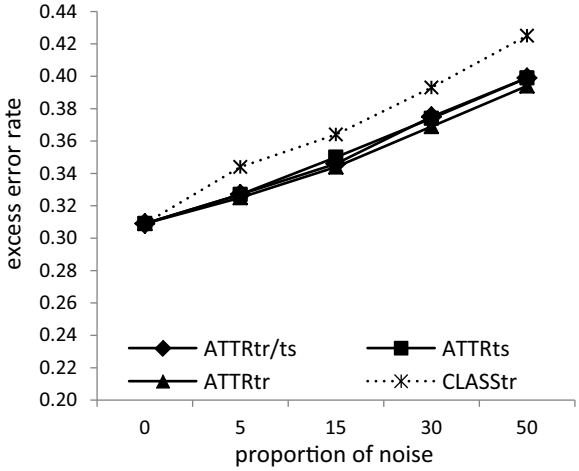

Fig. 6. Interaction between type of attribute and proportion of noise.

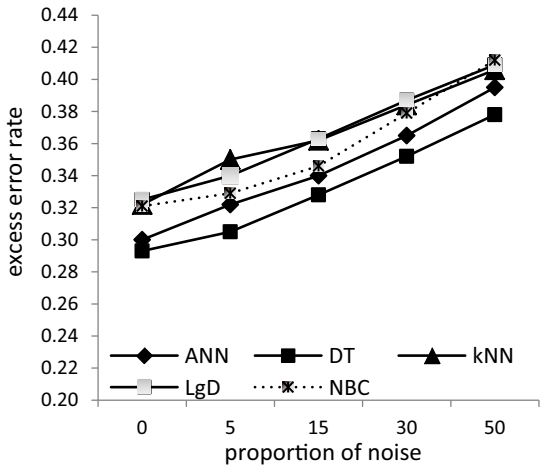

Fig. 7. Interaction between classifiers and proportion of noise.

All classifiers yield smaller error rate increases for attribute noise in the training set compared with classification noise. DT appears to handle all types of noise better than the other classifiers. However, the poor performance by all the classifiers when dealing with classification noise is noticeable.

The interaction effect between the types of attribute with noise and the different proportions of noise is displayed in Fig. 6. Once again, the results show accuracy rates increasing linearly to the growth of noisy attribute value level. A severe impact on classification accuracy is observed for classification noise with a lesser impact for attribute noise in the training set. Otherwise, all the classifiers degrade in performance with added noise.

For noise free data, Fig. 7 shows the classifiers as falling into two groups: those with higher accuracy rates (DT and ANN) and those with lower accuracy rates (NBC, kNN and LgD). Compared with the other classifiers, the performance of DT improves as the amount of noise increases. Otherwise, there appears to be no significant differences between the other classifiers at higher levels of noise.

Table 1
χ^2 values between attributes and class (loan payments dataset)

χ^2	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	A_9	A_{10}	A_{11}	A_{12}	A_{13}	A_{14}	A_{15}	A_{16}	A_{17}	A_{18}
Class	83.13	147.62	33.33	35.63	225.00	6.00	21.43	117.14	47.62	83.33	63.94	44.61	63.97	72.2	16.34	21.23	66.99	261.54

Table 2
χ^2 values between attributes and class (texas banks dataset)

χ^2	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	A_9	A_{10}	A_{11}	A_{12}	A_{13}	A_{14}	A_{15}	A_{16}	A_{17}	A_{18}	A_{19}
Class	72.2	166.9	48.71	58.84	60.79	36.66	11.45	22.06	44.8	42.67	128.51	5.23	119.14	11.69	14.48	9.57	14.69	56.00	34.13

Table 3
χ^2 values between attributes and class (australian credit approval)

χ^2	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	A_9	A_{10}	A_{11}	A_{12}	A_{13}	A_{14}	A_{15}
Class	62.95	65.87	159.32	61.24	0.54	4.57	71.89	69.45	183.87	0.40	32.28	120.06	1.81	67.22	11.39

Table 4
χ^2 values between attributes and class (german credit dataset)

χ^2	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	A_9	A_{10}	A_{11}	A_{12}	A_{13}	A_{14}	A_{15}	A_{16}	A_{17}	A_{18}	A_{19}	A_{20}
Class	79.71	26.83	195.76	153.60	20.16	125.60	120.83	251.71	52.31	9.39	211.25	18.40	114.53	116.12	58.9	22.06	11.49	131.25	8.89	85.17

4.2.2.2. Results for individual datasets

For any dataset, we execute the χ^2 test (described in Section 3) between each attribute and class, and provide the results in Tables 1–4.

After we compare the results from Figs 1–6 and the corresponding χ^2 values in Tables 1–4, some interesting conclusions can be drawn as follows:

1. The noise of different attributes has difierent impact with the system performance. The impact of the attribute noise critically depends on the dependence between the attribute and class.
2. Given an attribute A_i and class C, the higher the correlation between A_i and C, the more impact could be found from this attribute (A_i), if we introduce noise into A_i. As demonstrated in the Loan payments (Table 1), where attribute 18 has the highest χ^2 value with C, adding noise into attribute 18 has the largest impact (in the term of the accuracy decrease) in comparison with all other attributes (when the same noise level is added to each attribute). For the Texas bank dataset, the highest χ^2 value is for attribute 2; for the Australian credit and German credit datasets, the highest value for χ^2 re for attribute 9 and attribute 8, respectively.
3. If attribute A_i has very small correlation with class (or not at all), introducing noise into A_i usually has not much impact with the system performance. As demonstrated in the Loan payments (Table 1), where attributes 6, 7 and 15 have very small χ^2 vs with the class, all these three attributes are independent with the class C. Adding noise into these three attributes have no impact with the system performances, i.e., no matt how much noise has been introduced into these attributes, it would not affect the classification accuracy. Adding noise into attributes 7, 12, 14 and 16 for the Texas banks dataset) has no impact on the system performance. The same could be said for attribute 5, 6, 10, 13 and 15 (for Australian credit dataset) and attributes 5, 10, 12, 16, 17 and 19 (for German credit dataset).

The above conclusions indicate that the impact of noise from different attributevaries significantly with the classification accuracy, determined by the correlation between the corresponding attribute and class. This implies that when handling attribute noise, it's not necessary to dwith all attributes, and we may focus on some noise sensitive attributes only.

We shall now present experimental results that illustrate specific deviations from the overall results of the effectiveness of the five classifiers in terms of attribute and class noise. The rests report error rate of each method and are analysed from the perspective of each of the input data characteristics.

Loan payments

From Fig. 5, the overall best performance for attribute noise is by DT, with LgD achieving lower accuracy rates. The appears to be no significant differences in performances by all classifiers when dealing with attribute noise in both the training and test sets and the attribute noise in only the training set. More impact on classification accuracy is observed on attribute noise in only the test sets. For class noise, the results show DT (once again) yielding the best performances. Similarly, LgD is once again the less effective for this condition. In this class noise, NBC becomes more effective as the percentage of noise increases.

For this smallest dataset in our experiments, it appears that DT handles this dataset quite well for all types of noise. The good performance of DT is not surprising given its pruning strategy that eliminates noise in data. Also, the superior performance of kNN to NBC (for handling attribute noise) is rather surprising given the fact that its performance depends heavily on the value of k which also determines its efficiency for dealing with noisy data. In addition, this kind of data has purely numerical attributes, which DT would normally handle quite well.

Texas bank

From Table 6 it appears that for all type of attribute noise, DT performs better than the other classifiers,

followed by ANN, kNN, NBC and LgD, respectively. This is the case for all levels of noise. Attribute noise on only the test set appears to be more severe compared to when noise is only on the training or training and test sets. For class noise, the results show DT, once again, achieving the best results (Table 6). Good performances are also observed for classifiers such as NBC and ANN.

For this kind of dataset (which is the second smallest datasets in our experiments), DT outperforms all the other classifiers when dealing with noise of any kind. It is also surprising to see that not only is DT effective as a method for handling attribute noise but even more effective for handling class noise, especially at higher levels of noise.

Australian credit

As can be seen from Table 7, the overall best performance for all types of attribute noise is by DT, with ANN as a serious competitor. NBC performs as well as kNN while LgD yields the biggest error rates at all levels of noise. We further present results for the class noise suite which suggests an increase in error rates in most cases compared with the three types of attribute noise (Table 7). DT and ANN yield the best performances with no clear 'winner' between the two classifiers. Another good performance is by kNN which outperforms both LgD and NBC.

DT has a very good performance for the second biggest dataset in our experiments which contains many nominal attributes. This is the case when noise is in both attributes and class. Also, the performance of DT seems to be better on average when noise is are distributed among both the training and test sets. The poor performance of NBC to LgD for dealing with class noise is surprising given that LgD performed badly for all the other datasets. The poor performance of kNN is expected due to its heavy dependence on the type of distance measure used and the determination of the value of k for it to be more effective.

German credit

For the German credit data problem, the effects of noises on classification accuracy are summarised in Table 8 in the Appendix.

In terms of tolerating attribute noise, ANN performs slightly better than the other classifiers with DT being the least effective, especially when noise is in both the training and test sets. This is the case at all levels of noise. For class noise, kNN yields the best performance followed by ANN, DT, NBC and LgD, respectively. The impact of noise of classification accuracy appears to be more severe when it is on class compared to when it is on attributes.

For this kind of dataset it appears that ANN handles this dataset quite well, especially for attribute noise. Although the performances of ANN and DT (as classifiers for dealing with attribute noise) are not significantly different, its poor performance is still rather surprising given its pruning strategy that eliminates noise in data. However, as expected, DT exhibit higher accuracy rates when dealing with class noise, followed by ANN, kNN, NBC and LgD, respectively. In addition, this kind of data has mostly nominal attributes, of which you would expect DT to handle quite well.

5. Discussion and conclusions

The major focus of our research was to demonstrate the robustness of classifiers in credit risk noisy environments and further show how different types of noise impacts on predictive accuracy.

The comparison with current classifiers yielded a few interesting results.

This study shows that as the level of noise increases, the ability of the classifier to retain the baseline accuracy decreases with DT and ANN having higher accuracies for corrupted data with their accuracies

deteriorating lesser in comparison with classifiers such as NBC or kNN. In fact, NBC, kNN and LgD show bad results, being less tolerant to classification noise, especially at higher levels. In most cases, the decrease in accuracy is linear with respect to the increase of noise level. The better performance of DT compared to the other techniques can be attributed to the pruning strategy the algorithm has. The pruning strategy is designed to reduce the chance that the tree is overfitting to noise, especially in the training data.

While this study confirms statistical insight on the importance of the level of noise, it also reveals that the distribution of noise among the predictors for training and testing (classification) plays an equally (if not more) decisive role, depending on the amount of noise. Our study concludes that the existence of classification noise can bring lower predictive accuracy for supervised learning while higher accuracy rates are achieved by classifiers when noise in attributes is only on the training data. In other words, classification noise appears to be the dominant data quality factor in credit risk modelling, affecting LgD and kNN more profoundly than DT or ANN. Thus, removing noisy instances will significantly improve the performance of a classifier learned from noisy environments.

In addition, the behaviour of different classifiers varies from one dataset to another. Moreover, as illustrated by Texas banks data, attribute noise (especially on test data) can sometimes have some impact on predictive accuracy more than classification noise although the difference were found not to be statistically significant. Nonetheless, all classifiers were able to handle the bigger datasets better compared to smaller datasets as highlighted by the bigger error rates (for small datasets) and smaller error rates (for bigger datasets).

So far, we have restricted our experiments to only five classifiers. It would be interesting to carry out a comparative study of the top ten classifiers in data mining which can also handle noisy data. Furthermore, it would be possible to explore the different patterns and levels of noise in data.

Our research work was applied on only four datasets. This work could be extended by considering a more detailed simulation study using much more balanced types of datasets required to understand the merits of classifiers, especially larger datasets with noise on both categorical and numerical attributes.

We leave the above issues to be investigated in the future.

In sum, this paper provides the beginnings of a better understanding of the relative strengths and weaknesses of supervised learning techniques for handling noisy data. It is hoped that it will motivate future theoretical and empirical investigations into noise data and supervised learning, and perhaps reassure those who are uneasy regarding the use of high quality data in prediction.

Acknowledgements

The work was funded by the Department of Electrical and Electronic Engineering Science at the University of Johannesburg in South Africa. The author would like to thank Chris Jones and Wray Buntine for helpful discussions and useful comments.

References

[1] A.R. Abdel-Khalik and El-Sheshai, Information choice and utilization in an experiment on default prediction, *Journal of Accounting Research* (Autumn 1980), 325–342.
[2] D.W. Aha, D. Kibler and M.K. Albert, Instance-based learning algorithms, *Machine Learning* **24** (1991), 173–202.
[3] E. Altman, Financial ratios, discriminant analysis and the prediction of corporate bankruptcy, *Journal of Finance* **23** (1968), 589–609.

[4] B. Baesens, R. Setiono, Ch. Mues and J. Vanthienen, Using neural network rule extraction and decision tables for credit risk evaluation, *Management Science* **49** (2003), 312–239.
[5] G. Batista and M.C. Monard, An analysis of four missing data treatment methods for supervised learning, *Applied Artificial Intelligence* **17** (2003), 519–533.
[6] C.L. Blake and C.J. Mertz, UCI repository of machine learning databases, University of California, Department of Information and Computer Science, Irvine, 1998. http://www.ics.uci.edu/∼mlearn/MLRepository.html (22 October 2011).
[7] L. Breiman, J. Friedman, R. Olshen and C. Stone, *Classification and Regression Trees*, Wadsworth, (1984).
[8] C.E. Brodley and M.A. Friedl, Identifying and eliminating mislabelled training instances, in: *Proceedings of the 13*$^{\text{th}}$ *National Conference on Artificial Intelligence* Portland, OR. AAAI Press, (1996), 799–805.
[9] C.E. Brodley and M.A. Friedl, Improving automated land cover mapping by identifying and eliminating mislabelled observations from training data, in: *Proceedings of the International Geoscience and Remote Sensing Symposium*, Lincoln, NB, **11** (1996), 1382–1384.
[10] C.E. Brodley and M.A. Friedl, Identifying mislabeled training data, *Journal of Artificial Intelligence Research* **11** (1999), 131–167.
[11] BIS, *International Convergence of Capital Measurement and Capital Standards: A Revised Framework*, Basel Committee on banking Supervision, bank of International Settlements, 2004.
[12] D.R. Cox, Some procedures associated with the logistic qualitative response curve, in: *Research papers in Statistics: Festschrift for J Neyman*, F.N. David, ed., Wiley, New York, (1966), 55–71.
[13] S. Chatterjee and S. Barcun, A nonparametric approach to credit screening, *Journal of American Statistical Association* **65** (1970), 50–154.
[14] S.P. Curram and J. Mingers, Neural networks, decision tree induction and discriminant analysis: An empirical comparison, *Journal of the Operational Research Society* **45**(4) (1994), 440–450.
[15] R.H. Davis, D.R.H. Edelman and D.B. Gammerman, Machine learning algorithms for credit card applications, *IMA Journal of Mathematics Applied in Business and Industry* **4** (1988), 43–51.
[16] N.E. Day and D.F. Kerridge, A general maximum likelihood discriminant, *Biometrics* **23** (1967), 313–323.
[17] T.G. Dietterich, An experimental comparison of three methods for constructing ensemble of decision trees: Bagging, boosting and randomization, *Machine Learning*, (1999), 1–22.
[18] P. Domingos and M. Pazzani, Beyond independence: Conditions for the optimality of the simple Bayesian classifier, in: *Proceedings of the 13*$^{\text{th}}$ *International Conference on Machine Learning*, Bari, Italy, (1996), 105–112.
[19] R. Duda and P. Hart, *Pattern Classification and Scene Analysis*, New York: John Wiley, 1973.
[20] D. Feldman and S. Gross, Mortgage default: Classification tree analysis, *Journal of Real Estate Finance and Economics* **30** (2005), 369–396.
[21] FICO, FICO score, http://www.fico.com/en/FIResourcesLibrary/FICO_Score_1655PS.pdf, (Accessed on 20 June 2012).
[22] R. Florez-Lopez, Effects of missing data in credit risk scoring: A comparative analysis of methods to achieve robustness in the absence of sufficient data, *Journal of Operational Research Society*, (2010), 486–501.
[23] D.J. Hand, *Construction and Assessment of Classification Rules*, New York: Wiley, 1997.
[24] D.J. Hand and V. Vinciotti, Choosing k for two-class nearest neighbour classifiers with unbalanced classes, *Pattern Recognition Letters* **24** (2003), 1555–1562.
[25] W.E. Henley and D.J. Hand, A k-nearest neighbour classifier for assessing consumer credit risk, *Statistician* **44** (1996), 77–95.
[26] D.W. Hosmer and S. Lameshow, *Applied Logistic Regression*, New York: Wiley, 1989.
[27] C.C. Holmes and N.M. Adams, A probabilistic nearest neighbour method for statistical pattern recognition, *Journal of the Royal Statistical Society*, Series B **64** (2002), 295–306.
[28] M.J. Islam, Q.M.J. Wu, M. Ahmadi and M.A. Sid-Ahmed, Investigating the performance of Naive-Bayes classifiers and K-Nearest neighbor classifiers, *International Conference on Convergence Information Technology (ICCIT (2007))*, (2007), 1541–1546.
[29] G.H. John, Robust decision trees: Removing outliers from databases, in: *Proceedings of the 1*st *International Conference on Knowledge Discovery and Data Mining*, (1995), 174–179.
[30] E. Kalapanidas, N. Avouris, M. Cracium and D. Neagu, Machine learning algorithms: A study on noise sensitivity, in: *Proc 1st Balcan Conference in Informatics*, Y. Manolopoulos and P. Spirakis, eds, Thessaloniki, (November 2003), 356–365.
[31] A. Kalousis and M. Hilario, Supervised knowledge discovery from incomplete data, in: *Proceedings of the 2*nd *International Conference on Data Mining 2000*, Cambridge, England, WIT Press, July 2000.
[32] R.E. Kirk, *Experimental design* (2$^{\text{nd}}$ Ed.), Monterey, CA: Brooks, Cole Publishing Company, 1982.
[33] I. Kononenko, Semi-naïve Bayesian classifier, in: *Proceedings of European Conference on Artificial Intelligence* (1991), 206–219.
[34] E. Lawrence and N. Arshadi, A multinomial logit analysis of problem loan resolution choices in banking, *Journal of Money, Credit and Banking* **27** (1995), 202–216.

[35] K. Lakshminarayan, S.A. Harp and T. Samad, Imputation of missing data in industrial databases, *Applied Intelligence* **11** (1999), 259–275.
[36] P. Langley and S. Sage, Induction of selective Bayesian classifiers, in: *Proc Conf on Uncertainty in AI*, Morgan Kauffmann, 1994.
[37] R.J.A. Little and D.B. Rubin, *Statistical Analysis with missing data*, New York: Wiley, 1987.
[38] J.S. Long, *Regression Models for Categorical and Limited Dependent Variables Advanced Quantitative Techniques in the Social Sciences Number 7*, Sage Publications: Thousand Oaks CA, 1998.
[39] P. McCullagh and J.A. Nelder, *Generalised Linear Models*, 2nd Edition, Chapman and Hall, London, England, 1990.
[40] S. Menard, *Applied Logistic Regression* Sage Publications, Inc, Thousand Oaks, CA, 1995.
[41] W.F. Messier and J.V. Hansen, Inducing rules for expert system development: An example using default and bankruptcy data, *Management Science* **34**(12) (1988), 1403–1415.
[42] MINITAB, *Statistical Software for Windows 90*. MINITAB, Inc., PA, USA, 2002.
[43] T.M. Mitchell, *Machine Learning*, McGraw Hill, 1997.
[44] M.M. Monago and Y. Kodratoff, Noise and knowledge acquisition, in: *IJCAI-87*, J. McDermott, ed., Kaufmann, CA, (1987), 348–354
[45] J.P. Pinder, Decision analysis using multinomial logit models: Mortgage portfolio valuation, *Journal of Economics and Business* **48** (1996), 66–77.
[46] S. Piramuthu, Financial credit-risk evaluation with neural and neurofuzzy systems, *European Journal of Operational Research* **112** (1999), 310–321.
[47] J.R. Quinlan, Induction of decision trees, *Machine Learning* **1**(1) (1986), 81–106.
[48] J.R. Quinlan, Simplifying decision trees, *International Journal of Man – Machine Studies* **27** (1987), 221–234.
[49] J.R. Quinlan, Decision trees and decision making, *IEEE Transactions on Systems, Man and Cybernetics* **20**(2) (1990), 339–346.
[50] J.R. Quinlan, *C.4.5: Programs for machine learning*, Los Altos, California: Morgan Kauffman Publishers, INC, 1993.
[51] H. Ragavan, L. Rendell, M. Shaw and A. Tessmer, Complex concept acquisition through directed search & feature caching, and practical results in a financial domain, *Proceedings of the 13$^{\text{th}}$ International joint conference on Artificial Intelligence* (1993), 946–951.
[52] B.D. Ripley, *Pattern Recognition and Neural Networks*, Cambridge University Press, New York: John Wiley, 1992.
[53] B.D. Ripley, Neural networks and related methods for classification. *Journal of the Royal Statistical Society*, Series B **56**(3) (1994), 409–437.
[54] B.D. Ripley, *Pattern Recognition and Neural Networks*, Cambridge: Cambridge University Press, 1996.
[55] E. Rosenberg and A. Gleit, Quantitative methods in credit management: A survey, *Operations Research* **42** (1994), 589–613.
[56] D.E. Rumelhart, G.E. Hinton and R.J. Williams, Learning international representation by error propagation, in: *Parallel distributed processing: Explorations in the microstructure of cognition Volume 1: Foundations*, D.E. Rumelhart and J.L. McClelland, eds, Cambridge, MA: MIT Press, 1986.
[57] J.L. Schafer, *Analysis of Incomplete Multivariate Data*, Chapman and Hall, London, 1997.
[58] J.L. Schafer and J.W. Graham, Missing data: Our view of the state of the art, *Psychological Methods* **7**(2) (2002), 147–177.
[59] M.J. Shaw and J. Gentry, Inductive learning for risk classification, *IEEE Expert Intelligent Systems and their Applications* **5**(1) (1990), 47–53.
[60] P. Sibbersten, G. Stahl and C. Luedtke, Measuring model risk, *Journal of Risk Model Validation* **2**(4) (2009), 65–81.
[61] G. Stahl, E. Nill, B. Siehl and J. Wilsberg, Risk modelling and model risk – the IRBA case, *Working Paper*, Bonn Germany, 2008.
[62] K.Y. Tam and M.Y. Kiang, Managerial applications of neural networks: The case of bank failure predictions, *Management Science* **38**(7) (1992), 926–947.
[63] T.M. Therneau and E.J. Atkinson, An introduction to recursive partitioning using the RPART routines, Technical Report, Mayo Foundation, 1997.
[64] B. Twala, *Effective Techniques for Handling Incomplete Data Using Decision Trees*, Unpublished PhD thesis, Open University, Milton Keynes, UK, 2005.
[65] B. Twala, M.C. Jones and D.J. Hand, Good methods for coping with missing data in decision trees, *Pattern Recognition Letters* **29** (2008), 950–956.
[66] D. West, Neural network credit scoring models, *Computers & Operations Research* **27** (2000), 1131–1152.
[67] P. Winston, *Artificial Intelligence* Addison-Wesley, third edition Part II: Learning and regularity Recognition, 1992.
[68] I.H. Witten and E. Frank, Data mining: Practical machine learning tools and techniques, 2nd Edition, Morgan Kauffmann, Francisco, 2005.
[69] X. Zhu and X. Wu, Class noise vs. attribute noise: A quantitative study of their impacts, *Artificial Intelligence Review* **22**(3–4) (2004), 177–210.

Appendix

Table 5
Classification accuracy – Loan payments data

Noise level		0	5	15	30	50
Actual noise		0.0	2.5	11.5	26.3	31.8
ANN	$\text{ATTR}_{\text{TR/TS}}$	37.0	39.7	42.5	44.5	45.9
	ATTR_{TS}	37.0	39.3	41.7	43.2	46.8
	ATTR_{TR}	37.0	40.4	42.2	44.5	45.5
	CLASS_{TR}	37.0	39.9	41.3	44.2	46.4
DT	$\text{ATTR}_{\text{TR/TS}}$	34.6	36.9	38.7	40.1	42.2
	ATTR_{TS}	34.6	37.4	39.7	41.7	43.4
	ATTR_{TR}	34.6	36.1	37.2	39.7	41.8
	CLASS_{TR}	34.6	38.9	40.9	43.8	45.1
kNN	$\text{ATTR}_{\text{TR/TS}}$	38.6	40.9	42.1	43.6	45.4
	ATTR_{TS}	38.6	42.2	43.9	44.2	46.2
	ATTR_{TR}	38.6	40.5	41.7	42.6	44.0
	CLASS_{TR}	38.6	43.3	43.5	46.2	48.7
LgD	$\text{ATTR}_{\text{TR/TS}}$	41.2	42.7	43.6	45.2	47.2
	ATTR_{TS}	41.2	44.3	44.6	45.9	48.9
	ATTR_{TR}	41.2	42.4	42.9	45.3	47.3
	CLASS_{TR}	41.2	43.5	44.0	46.2	48.9
NBC	$\text{ATTR}_{\text{TR/TS}}$	39.8	40.7	41.3	43.8	46.4
	ATTR_{TS}	39.8	41.0	43.1	45.4	46.7
	ATTR_{TR}	39.8	41.3	43.3	45.4	47.9
	CLASS_{TR}	39.8	42.7	43.3	46.9	48.0

Table 6
Classification accuracy – Texas banks data

Noise level		0	5	15	30	50
Actual noise		0.0	2.5	11.5	26.3	31.8
ANN	$\text{ATTR}_{\text{TR/TS}}$	32.7	35.6	37.4	39.6	38.8
	ATTR_{TS}	32.7	36.0	39.4	40.9	39.5
	ATTR_{TR}	32.7	33.3	36.1	37.1	40.1
	CLASS_{TR}	32.7	36.9	38.2	40.7	43.3
DT	$\text{ATTR}_{\text{TR/TS}}$	30.9	33.3	36.0	37.4	39.5
	ATTR_{TS}	30.9	33.9	37.1	38.8	41.3
	ATTR_{TR}	30.9	30.6	35.8	35.7	37.4
	CLASS_{TR}	30.9	34.6	38.0	38.3	42.5
NBC	$\text{ATTR}_{\text{TR/TS}}$	34.6	36.1	37.1	40.1	43.5
	ATTR_{TS}	34.6	38.3	40.7	43.1	46.6
	ATTR_{TR}	34.6	38.9	40.1	42.0	44.3
	CLASS_{TR}	34.6	36.7	36.9	41.4	43.3
LgD	$\text{ATTR}_{\text{TR/TS}}$	34.7	35.1	37.8	40.7	42.2
	ATTR_{TS}	34.7	34.2	35.6	38.4	40.2
	ATTR_{TR}	34.7	40.1	40.8	43.1	45.9
	CLASS_{TR}	34.7	37.0	38.4	40.3	42.4
kNN	$\text{ATTR}_{\text{TR/TS}}$	33.1	34.3	34.3	37.1	39.3
	ATTR_{TS}	33.1	34.1	38.1	41.7	45.4
	ATTR_{TR}	33.1	35.1	36.1	43.0	46.3
	CLASS_{TR}	33.1	36.2	39.0	41.2	43.3

Table 7
Classification accuracy – Australian credit data

Noise level		0	5	15	30	50
Actual noise		0.0	2.5	11.5	26.3	31.8
ANN	$\text{ATTR}_{\text{TR/TS}}$	26.0	27.5	29.3	34.6	37.4
	ATTR_{TS}	26.0	26.8	28.1	32.1	35.2
	ATTR_{TR}	26.0	26.6	27.3	31.8	34.4
	CLASS_{TR}	26.0	29.0	31.2	37.1	41.0
DT	$\text{ATTR}_{\text{TR/TS}}$	24.6	26.7	29.0	33.8	36.8
	ATTR_{TS}	24.6	26.8	28.9	32.7	37.8
	ATTR_{TR}	24.6	26.0	27.1	31.1	34.6
	CLASS_{TR}	24.6	27.1	30.2	34.1	40.4
kNN	$\text{ATTR}_{\text{TR/TS}}$	30.9	32.6	34.1	38.4	42.2
	ATTR_{TS}	30.9	34.3	35.8	40.7	40.3
	ATTR_{TR}	30.9	36.7	37.5	39.4	41.0
	CLASS_{TR}	30.9	39.6	38.4	40.9	43.4
LgD	$\text{ATTR}_{\text{TR/TS}}$	31.2	34.3	36.7	42.4	46.4
	ATTR_{TS}	31.2	33.9	35.1	38.0	40.7
	ATTR_{TR}	31.2	33.3	34.2	40.2	38.4
	CLASS_{TR}	31.2	37.4	40.1	42.2	44.8
NBC	$\text{ATTR}_{\text{TR/TS}}$	28.4	30.1	32.6	39.6	40.9
	ATTR_{TS}	28.4	29.4	30.9	37.8	38.0
	ATTR_{TR}	28.4	28.6	29.8	35.6	36.1
	CLASS_{TR}	28.4	31.7	35.0	40.8	45.1

Table 8
Classification accuracy – German credit data

Noise level		0	5	15	30	50
Actual noise		0.0	2.5	11.5	26.3	31.8
ANN	$\text{ATTR}_{\text{TR/TS}}$	21.7	24.3	26.9	27.5	30.2
	ATTR_{TS}	21.7	23.5	25.5	26.5	28.9
	ATTR_{TR}	21.7	23.0	25.0	25.1	27.7
	CLASS_{TR}	21.7	25.7	28.6	28.7	32.8
DT	$\text{ATTR}_{\text{TR/TS}}$	22.9	27.2	30.7	33.7	35.7
	ATTR_{TS}	22.9	24.8	32.6	33.3	37.1
	ATTR_{TR}	22.9	24.2	28.8	31.8	35.7
	CLASS_{TR}	22.9	28.7	31.8	33.7	31.6
kNN	$\text{ATTR}_{\text{TR/TS}}$	26.2	28.7	29.0	30.6	33.7
	ATTR_{TS}	26.2	24.8	23.8	25.2	27.0
	ATTR_{TR}	26.2	26.2	26.3	29.1	31.5
	CLASS_{TR}	26.2	27.8	26.3	28.4	31.6
LgD	$\text{ATTR}_{\text{TR/TS}}$	27.5	30.0	32.2	34.7	35.1
	ATTR_{TS}	27.5	30.1	35.9	34.8	39.3
	ATTR_{TR}	27.5	28.5	31.6	32.0	34.9
	CLASS_{TR}	27.5	32.2	37.6	38.8	42.7
NBC	$\text{ATTR}_{\text{TR/TS}}$	25.7	26.4	26.6	27.6	34.0
	ATTR_{TS}	25.7	26.1	24.4	27.5	32.0
	ATTR_{TR}	25.7	24.8	24.0	25.5	31.6
	CLASS_{TR}	25.7	28.1	28.3	32.2	39.7

Table 9
Analysis of variance for significance tests for classifiers (significant effects only)

Source	Degrees of freedom	Sum of squares	Mean-square	F-ratio	$p-$value
Main effects:					
A	4	0.079	0.019	49.462	0.000
B	3	0.196	0.065	163.423	0.000
C	4	0.278	0.070	173.917	0.000
Two-way interactions:					
$A * B$	12	0.012	0.001	2.276	0.028
$B * C$	12	0.007	0.001	2.401	0.021

Note: A = classifiers (fixed factor with 5 levels); B = type of attribute with noise and class noise (fixed factor with 4 levels); C = proportions of noise (fixed factor with 4 levels).