Dynamic Credit Scoring Using Payment Prediction

Raymond Sunardi Oetama

A dissertation submitted to
Auckland University of Technology
in fulfilment of the requirements for the degree of
Master of Computer and Information Sciences

2007

Computing and Mathematical Sciences at AUT

Primary Supervisor: Dr Russel Pears

Table of Contents

Index of Tables

Index of Figures

Index of Equations

Attestation of Authorship

“I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person (except where explicitly defined in the acknowledgements), nor material which to a substantial extent has been submitted for the award of any other degree or diploma of a university or other institution of higher learning."

Yours sincerely,

(Raymond Sunardi Oetama)

Abstract

Credit scoring is a common tool used by lenders in credit risk management. However, recent credit scoring methods are error-prone. Failures from credit scoring will significantly affect the next process, which is payment collection from customers. Bad customers, who are incorrectly approved by credit scoring, end up making payments that are overdue.

In this dissertation, we propose a solution for pre-empting overdue payment as well as improving credit scoring performance. Firstly, we utilize data mining algorithms including Logistic Regression, C4.5, and Bayesian Network to construct payment predictions that can quickly find overdue payments in advance. By utilizing payment prediction, customers who may make overdue payments will be known by the lender earlier. As a result, the lender can proactively approach such customers to pay their payments on schedule. The second solution is to define a refined scoring model that will use feedback from the payment prediction models to improve the initial credit scoring mechanism. The payment prediction result will give information to review the combinations of current credit scoring parameters that work inappropriately. By updating the current credit scoring parameters, the performance of credit scoring is expected to increase significantly. As a result, this mechanism will create a dynamic credit scoring model.

We also investigate the impact of the imbalanced data problem on the payment prediction process. We employ data segmentation as a tool to overcome the problem of imbalanced data. By using a novel technique of data segmentation, which we call Majority Bad Payment Segments (MBPS), learning bad payments become much easier. The results of our experiments show that payment prediction based on MBPS produces much higher performance when compared to conventional methods of dealing with imbalanced data. We perform extensive experimentation and evaluation with a variety of metrics such as Hit Rates, Cost Coverage, F-measure, and the Area under Curve measure.

Acknowledgment

It was really not easy to accomplish writing this dissertation. I had to work hard to find a suitable solution for both credit scoring and payment collection problems. Therefore, first of all, I would like to acknowledge and thank Dr Russel Pears, my supervisor, who strongly supported me along the whole difficult time in finishing this dissertation.

I would like to thank many friends from Indonesia, my country, who sent me their data that is being used in this research. Although I am unable to cite all their names as per their request, however, without their help, I would never have been able to finish this research.

I would especially like to thank my wife Ni Made Sri Utari, my daughters Adella Charlene Oetama and Pamella Cathryn who never stopped to support me morally and through uncountable prayers to help me be a strong father all the time and to complete this dissertation from the beginning to the end.

Chapter 1: Introduction

Purchase of expensive things such as motor cycles or cars needs a large amount of cash. Today, instead of cash, people can pay in instalments over a period of time spanning three months, six months, one year, two years, etc. Credit payments enable such types of payments. Credit payments are facilitated by lending companies such as finance companies or banks.

However, credit payment is not automatically granted to all customers. Only some of them can be approved depending on the lender's criteria. All such criteria assess the probability that the loan will be repaid back in the future by the customer. Typically, lender criteria can be divided into five categories, which are commonly called ***5C***. Firstly, lenders will analyse customer ***characteristics,*** meaning who the customer is. The second category is customer ***capacity*** to repay the loan. Customer capacities typically correspond to the monthly excessive income they may have. The third category is ***collateral,*** which are other valuable assets that can be pledged for repayment of the loan. For instances, cars, properties, etc. Next category is customer ***capital***, which include individual investments, insurances, etc. The last category is ***condition*****,** which cover other related situational facts such as market condition, social condition, etc.

Furthermore, customers will be asked to fill credit application forms that contain lender credit criteria and to provide supporting documentation such as photo id, the last three month bank account statements, etc. After the customers have filed an application, a credit analyst officer will assess the credit worthiness of the customer concerned. If all lender criteria are fulfilled by the customer then the credit application will be approved.

However, due to rapid business expansion of credit products such as consumer credits, property mortgages, etc, the manual approval process tends to overwhelm credit analysts with too many credit applications (Abdou, Masry, & Pointon, 2007). Crook et al. (2006) shows that between 1970 and 2005, consumer credit outstanding balance in US grew by 231% with a dramatic growth of 705% on property mortgages. Therefore, the manual credit analysis process is enhanced through the use of statistical methods (Servigny & Renault, 2004). A typical method of statistical approval method is credit scoring. Credit Scoring is defined as a set of tools that help to determine prospect for loan approval (Johnson, 2006).

Besides, after the credit applications have been approved, lenders will inform customers that their credit applications have been granted. This will generally lead to a customer signing a contract. On the contract, a payment schedule informs the customer of the amount and due date of payments the customer must repay the lender at particular points in time.

The majority of customers make their payments on schedule, but some customers do make late payments. Payments that are paid after the due date are called overdue payments. Collecting overdue payments may not be easy depending on the willingness of customers to pay. If customers still want to pay their overdue payments, lenders may arrange some methods to help them. In other cases, customers simply refuse to make their payments. As a result, such customers will create collection problems. Overdue payments occur since credit scoring imperfectly filters some bad customers. We identify two related problems that will be addressed in this dissertation. Firstly, credit scoring is imperfect causing overdue payments. Secondly, overdue payments directly give rise to payment collection problems.

The objective of this dissertation will be to provide solutions for both credit scoring and collection problems. The proposed solution is essentially a payment prediction of all overdue payments at the next payment periods in order to find all potential overdue payments in advance. As a result, some proactive actions can be taken to pre-empt overdue payments. Since all credit parameters are involved in building payment prediction, payment prediction results will show combinations of all credit scoring parameters that cover overdue payments. Such information can be utilized to improve the current credit scoring. This mechanism will create a dynamic credit scoring system.

This dissertation is organized in five chapters. This chapter has given an introduction to credit scoring and the overdue payment problem. The second chapter is a literature review that essentially consists of analysis of previous studies on credit scoring problems and their solutions. In the third chapter, a suitable methodology will be presented for the research. The next chapter will discuss experimental results of the proposed solutions. Finally, the last chapter summarises the research carried out and gives some directions for future research.

Chapter 2: Literature Review

2.1. Introduction

This chapter is starts by investigating credit scoring performance. We find that credit scoring performance is imperfect as it is incapable of rejecting all bad customers. Such customers will create problems in payment collection. Thereafter, the discussion is centred on an examination of solutions for both credit scoring problems and payment collection problems. Solutions for those problems comprise of algorithmic approaches and data centric approaches. Finally, we also review some appropriate metrics to evaluate algorithmic performance.

2.2. Credit Scoring

Models of credit scoring comprise of credit scoring parameters and mathematical functions to calculate credit scores based on such parameters. Credit scoring parameters actually represents lenders' criteria. Finlay (2006) gives examples of credit scoring parameters for personal loans; applicant gross income, time in employment, car ownership , etc. Credit scoring models are based on credit scoring objectives, algorithms and data sets.

Generally, the credit scoring objective is to assess credit worthiness of a customer. However, definitions of credit worthiness vary according to the credit scoring research arena. Firstly, large body of researchers focus on ***behavioural scoring.*** Behavioural scoring is to predict the odds of a customer being in default or not

(Thomas, Ho, & Scherer, 2001), i.e. being bad or good (T. S. Lee, Chiu, Lu, & Chen, 2002), Another credit scoring research objective is ***bankruptcy scoring***, where the study objectives are mainly to predict the likelihood of an individual customer declaring himself or herself bankrupt (Sun & Senoy, 2006). A further form is ***profit scoring*** (Crook et al., 2006), where lenders will calculate profitability of customers to the lender instead of calculating his or her credit risk. Finally, we find that other researchers pay more attention to predicting financial status such as outstanding balance, called ***loan projection scoring*** in this dissertation. Financial state classification may differ amongst lenders. Avery et al. (1996) divide their financial state classification into periods covering 1-30, 31-60, 61-90, and 91-120 overdue days. However, Smith et al apply a different form that comprises of five states: current (payment on schedule), 30 to 89 overdue days, more than 90 overdue days, defaulted, and paid off (Smith, Sanchez, & Lawrence, 1996). Boyes et al. (1989) simplify their classifications as repaid or defaulted.

One of the first algorithms used in credit scoring is a simple parametric statistic method called Linear Discriminant Analysis (LDA) (West, 2000). West explains that LDA has been criticised since covariance matrices of good and bad classes are significantly different. Thereafter, many researchers utilized other data mining algorithms, including Logistic Regression, Classification Tree, k-Nearest Neighbour, and Neural Network. Other applicable algorithms include Math Programming, Generic algorithm, Genetic algorithm, and Support Vector Machines (Crook et al., 2006), Naive Bayes (Baesens et al., 2003), and the Bayesian Network (Sun & Senoy, 2006).

Sources of credit scoring data sets vary enormously. A large body of researchers use German credit data sets from UCI Repository of Machine Learning Database (http://mlearn.ics.uci.edu/MLRepository.html) that contain 700 instances of good applicants and 300 instances of bad applicants. UCI data also provides Australian credit data sets comprising of 468 samples of good credits and 222 samples of bad credits. Other data are gathered from different sources such as annual expenditure and food survey from UK Government (Finlay, 2006), credit card applications from financial companies (Boyes et al., 1989), and personal loan datasets from banks in Egypt (Abdou et al., 2007), etc.

There are some advantages to utilizing credit scoring models. Firstly, the decision comes more quickly, accurately, impartially than with human assessment (Isaac, 2006). Secondly, it is utilized to ensure objective, consistent and manageable decision making (Laferty, 2006). Laferty adds other benefits of credit scoring such as automation capability using an IT platform, unbiased and consistent assessments because they are based on data analysis, and management control because it allows management to control and manage the level of risk.

2.3. Credit Scoring Problems

Although credit scoring enables lenders to accelerate the credit approval process, in fact, credit scoring does not perfectly identify all bad customers. A credit scoring model from Tsaih et al 's study shows an error rate of 20% (Tsaih, Liu, Liu, & Lien, 2004). A proposed credit scoring model from Li et al. (2004) shows better performance from their current credit scoring model, but it is still shows an error rate of 7.3%. Some researchers apply credit scoring to mobile phone users. They report 9.75% of trusted customer's bills are not paid whilst 11.38% of non-trusted customers pay on time (z. Li, Xu, & Xu, 2004). Another study result shows a total good applicant hit rate is 76.3% and total bad applicant hit rate of 84.6% (Zekic-Susac, Sarlija, & Bensic, 2004), but such figures are relatively far from an ideal hit rate of 100%.

Failures from credit scoring will significantly impact the next process, which is payment collection from bad customers. Bad customers fail to make their payments on schedule. In 1991, overdue payments in real estate products of Manufacturers Hanover was US$3.5 billions (West, 2000). If lenders are unable to recover their money from those bad customers, lenders incur huge losses, impacting on the economic performance of the companies involved. West highlights that this company lost US$385 million in the same year.

A relationship between failures of credit scoring and overdue payments is found in the Avery et al (1996) study. Generally, a higher credit score of will reflect a higher creditworthiness of a customer. Those customers who are scored higher are expected to pay their payments on schedule better than lower scored customers. However, Avery et al find some surprising results. They find that the largest portion of overdue payments comes from the higher end of the credit score range. By using mortgage data covering October 1993-June 1994, Avery et al show from a total of 109,433 customers, 417 have payments overdue by at least 30 days. Most of those overdue payments (60.9%) are from the high end of the credit score range, 21.8% from middle range, and the rest, comprising 17.3%, from the low range.

Most previous researchers overlook overdue payments since their work is focused on the improvement of credit scoring performance. Moreover, there is no proactive action from lenders to pre-empt such overdue payments. Typically lenders can find all overdue payments when they generate overdue payment reports. Since scheduled payments are generally due monthly, typically overdue payment reports will be categorized into 1-30, 31-60, 61-90 and 91-120 days overdue (Avery et al., 1996). For example if the due date is 31 January 2007 and a customer pays on 15 February 2007, the payment is about 15 days overdue and is categorized as 1-30 days overdue. By using overdue reports, lenders will know customers who fail to make their payments on time. Thereafter, lenders will take steps to collect overdue payments from such customers. This collection process itself is a problem because data collection occurs after the payment is actually due.

Many researchers propose solutions to improve the current credit scoring performance. Such solutions can be divided into two approaches, comprising of algorithmic approaches and data centric approaches. Algorithmic approaches will be discussed in section 2.4., whilst data approaches will covered in section 2.5.

2.4. Algorithmic Approaches

We find that there is no single best algorithm across different data domains. An algorithm may be the best on some particular datasets, but it will perform worse than the other algorithms on different datasets. Srinivasan and Kim (1987) show that the Decision Tree has the best performance on their dataset, but West (2000) show that Neural Networks perform better than Decision Trees on their dataset. In contrast, Desai et al. (1996) report when predicting good and bad customers, Logistic Regression outperforms Neural Network on their dataset. However, Ong et al. (2005) show that the best algorithm on their dataset is not Decision Tree, Neural network, or Logistic Regression, but Genetic Programming (Koza, 1992).

Since there is no single algorithm that performs best for all datasets, we conclude that our research will fare better if we can find the best algorithm for our specific dataset. Therefore it is required to involve a number of different algorithms in order to find the best algorithm. Justification for the inclusion of such algorithms will be given in the methodology chapter.

2.5. Data Centric Approaches

Credit scoring problems are viewed as imbalanced data problems (Chawla, Japkowicz, & Kotcz, 2004). The imbalance problems occurs when one class has more examples than the others, thus reducing classification performance on the minority classes (Weiss, 2004). The class that contains the most amounts of examples is called the majority class whilst the others are called minority classes. A study from Weng and Poon (2006) shows the effect of the imbalanced data problem on classification performance. Initially, their original dataset is virtually balanced with a ratio close to 1:1 between classes. After a removal of 20% of instances from the minority class, their classifier accuracy rate dropped from 95% to 94%. After an extreme reduction of 95% of minority data, the classifier accuracy significantly decreased to 72%. In 2007, (Wei, Li, & Chen, 2007) study reports on a two class imbalanced dataset. The total number of record in their dataset is 5000 records that consist of 4185 good class and 815 bad class records. Huang, Hung, and Jiau (2006) divide their creditworthiness data into 3 classes that consists class 1, class 2 and class 3 with a ratio of 19:15:66. Class 3 is viewed as the majority class whilst others are minority classes.

Common data solutions for imbalanced datasets are under sampling and over sampling. In under sampling, some instances of the majority class are excluded so that the ratio between majority and minority class are more balanced with respect to each other. In contrast, in applying over sampling, data is balanced by adding some more instances to the minority classes. There is one popular data over sampling method for numerical data called SMOTE (Synthetic Minority Over-sampling Technique). SMOTE applies a random generator to create new instances of minority class that lie on the boundary between majority class instances and minority class instances (Chawla, Bowyer, Hall, & Kegelmeyer, 2002).

Weiss reveals that data segmentation is also a solution to the imbalanced data problem. There are some reasons to using segmentation. By applying data segmentation, researchers can pay more attention on target segments and ignore other segments (Shihab, Al-Nuaimy, Huang, & Eriksen, 2003). Segmentation can also be utilized to compare data amongst segments (Rimey & Cohen, 1988).

Techniques for data segmentation for credit scoring problems vary amongst researchers. Lee and Zhang (2003) divided their datasets into *k* segments. Their solution was to create a score model using Logistic Regression for each segment since their segmentation reflects heterogeneity across the population. Hsieh (2004) aligns his credit scoring data with marketing strategies by segmenting data based on their repayment behaviour using a Neural Network. The outcome of their research is the creation of marketing incentive programs for some high-value customers.

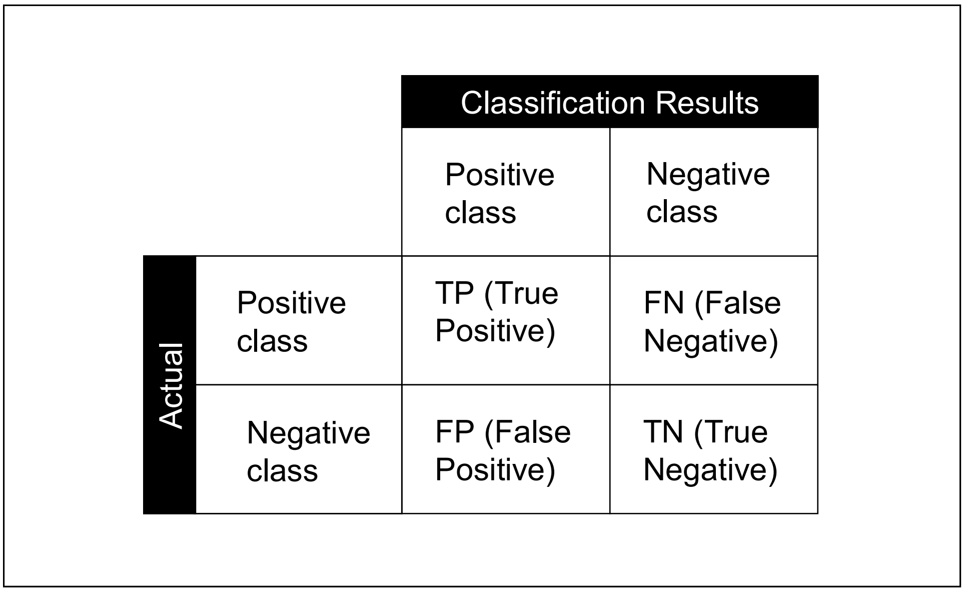
In comparison with both under sampling and over sampling methods, data segmentation has some advantages. The Lee and Zhang study highlights advantages such as better adaptability since the changes arising from local condition may affect only certain segments and thus not require global changes to be made. Whilst for sampling methods, such changes may affect the entire model. As a result, both under sampling and over sampling must rebuild their models in such situations. Learning from Hsieh's model, segmentation can be aligned to the business needs such as marketing customer segmentation, whilst neither under sampling nor over sampling methods supports such advantages.

Furthermore, Weiss also reveals that an alternative method of learning minority classes on imbalanced data set is one-class learning. The characteristics of a rare class are recognized by learning only on the minority class rather than involving the majority class. Weiss give an example from Brute's study about failures (rare class) in Boeing manufacturer. Brute focuses only on rules that predict failures to obtain better result about the failures. However, according to Chawla et al (2004), one-class learning can be performed if only there is a clean split between majority class and minority classes. We believe that most bad customers share some characteristics with good customers. Good customers in this case cannot be cleanly separated from bad customers or ignored since some characteristics belong to both good customers and bad customers. Segmentation is more applicable since it will not separate the classes, but will transform all data into segments and data mining will be performed only on segments that contain the majority of instances from the minority class (Weiss, 2004).

Amongst imbalance data problems, we prefer to utilize segmentation as it can be utilized to learn about overdue payments and it is more applicable than one-class learning. A complete justification as well as a method of performing segmentation will be given in the next chapter.

2.6. Evaluation Metrics

Evaluation metrics are very important since they are utilized to analyse the algorithm performance. Evaluation metrics can be developed from the confusion matrix. As can be seen from Figure 2.1, for a two classes problem, the confusion matrix consists of four cells, which are ***True Positive*** (TP) meanings the minority instances that are classified correctly, ***False Positives*** (FP) correspond to majority instances that are classified incorrectly as minority class instances, ***False Negative***s (FN) where minority instances are classified wrongly as majority instances and ***True Negative*** (TN) where majority instances are classified correctly as minority instances. There some metrics that have been developed from the confusion matrix.



		Classification Results	
		Positive class	Negative class
Actual	Positive class	TP (True Positive)	FN (False Negative)
	Negative class	FP (False Positive)	TN (True Negative)

Figure 2.1: Confusion Matrix

Many credit scoring researches use ***accuracy*** in their study to measure classifier performance (Desai et al., 1996; Srinivasan & Kim, 1987). Accuracy represents percentage of examples that are correctly classified. Accuracy is calculated from the following expression:

$$ACCURACY = \frac{TP + TN}{TP + FP + FN + TN}$$

Equation 2.1

However, in regard to imbalanced data, Weiss (2004) points out that accuracy is not appropriate to measure overall performance since accuracy places more emphasis on the majority class. Han, Wang, and Mao (2005) explain that in imbalanced data classifications, many instances of majority class are predicted correctly, but many or all instances of minority class are predicted incorrectly. Since there are many more instances of the majority class than the minority class, accuracy is still high. Furthermore, Han et al strongly state that accuracy is not reliable for an imbalanced data set.

Another metric that can be derived from the confusion matrix is the **hit rate**. The hit rate corresponds to the percentage of the positive class that is correctly classified. In other studies hit rates are also called as the true positive rate, recall or sensitivity. Hit rate is calculated from the following equation:

$$HIT\,RATE = \frac{TP}{TP + FN}$$

Equation 2.2

Hit rate is appropriate for imbalanced data problems as it can be used to measure performance on either the majority or minority class. Zekic et al. (2004) applies hit rates for both the good applicant and bad applicant classes. For studies where the interest centres on the majority class it can be taken as the positive class, similarly, when the interest is in the minority class, then that class can be set as the positive class.

Another metric that can be derived from the confusion matrix is ***precision***. Precision corresponds to the accuracy on a given class. For the positive class, precision is given by the following equation:

$$PRECISION = \frac{TP}{TP + FP}$$

Equation 2.3

Many researchers utilize hit rates and precision when they apply ***F-measure***. F-measure (Rijsbergen, 1979) is the metric that can be used to observe both hit rate and precision at the same time. If hit rate and precision have the same weight, the F-measure is calculated as the following equation:

$$F - measure = \frac{2 \times HIT\,RATE \times PRECISION}{HIT\,RATE + PRECISION}$$

Equation 2.4

Ma and Cukic (2007) point out that F-measure is more appropriate than accuracy. A low accuracy on the minority class reflects a low F-measure whilst overall accuracy may still be high (Han et al., 2005).

Another popular metric for imbalanced data is derived from the ***ROC (Receiver Operating Characteristic).*** ROC is a two dimensional graph, which reflects true positive rate projection on its y-axis and the false positive rate on its x-axis where false positive rate = FP/ (TN+FP). Han et al adds (2005) that ROC actually reflects a trade-off between true positive rate and false positive rate.

One derivative metric from ROC is ***AUC or Area under ROC*** (Fawcett, 2005). AUC represents the probability of a randomly chosen majority class example against the probability of a randomly chosen minority. For random guessing, the AUC coefficient = 0.5. A good classifier will produce an AUC coefficient better than random guessing (>0.5). Weiss points out that AUC is more appropriate than accuracy since it is not biased against the minority class.

The basic rule of ***cost sensitive learning*** is explained well by Elkan (2001). Applying cost sensitive learning particularly for credit scoring can be found on Abdou et al (2007) following the guidance from West (2000). Abdou et al. define their cost function as:

$$Cost = C_{G-B} \times P_{G-B} \times \pi_G + C_{B-G} \times P_{B-G} \times \pi_B$$

Equation 2.5

where C_{G-B} is the cost of an actual good customer being predicted as a bad customer, P_{G-B} is probability of a good but being predicted as a bad customer, π_G is prior probability of a good customer, C_{B-G} is the cost of an actual bad being predicted as a good customer, with P_{B-G} as the probability of a bad customer being predicted as a good customer, π_B is prior probability of a bad customer.

Elkan argues that this cost function is useful to control failures to be as small as possible. The smaller the value of this cost function the better performance of the algorithm.

Thereafter, Abdou et al apply the prior probability of good customer as π_G= (TP+FN)/(TP+TF+FP+FN). Similarly, they also apply prior probability of bad customer as π_B = (TN+FP)/(TP+TF+FP+FN).

As West has observed, P_{G-B} is actually the false negative rate whilst P_{B-G} represents the false positive rate. Since P_{G-B} = FN/(TP+FN) and P_{B-G}= FP/(TN+FP), the cost function can be updated as:

$$Cost = C_{G-B} \times \frac{FN}{TP+FN} \times \frac{TP+FN}{TP+TF+FP+FN} + C_{B-G} \times \frac{FP}{TN+FP} \times \frac{TN+FP}{TP+TF+FP+FN}$$

Or can be simplified as

$$Cost = C_{G-B} \times \frac{FN}{TP+TF+FP+FN} + C_{B-G} \times \frac{FP}{TP+TF+FP+FN}$$

Equation 2.6

This function is valid if good customers are chosen as the positive class. If bad customers comprise the positive class then equation 2.6 becomes

$$Cost = C_{B-G} \times \frac{FN}{TP + TF + FP + FN} + C_{G-B} \times \frac{FP}{TP + TF + FP + FN}$$

Equation 2.7

since, for bad customers are the positive class, FN represents bad customers that are predicted as good customers whist FP represents good customers being predicted as bad customers.

Furthermore, for good customers as the positive class, researchers believe that the risk of predicting bad customers as good customers is higher than the risk of predicting good customers as bad customers (Nayak & Turvey, 1997). Nayak et al. explain that mistakes due to predicting bad customers as good customers will cost lost principal, lost interest, lost administration fee, legal fees, insurance coverage, and property taxes. However, the cost of predicting bad customers as good customers is typically unclear since it is not clear how much lost of expected profit is lost by rejecting a customer with good credit.

Following the findings of Hofmann (West, 2000), both West and Abdou et al. use a relative cost factor of 5 for predicting bad customers as good customers as opposed to predicting good customers as bad customers. However, the focus of our research is bad customers and the implications of this will be discussed in the research target segment of the next chapter.

2.7. Summary

We find that credit scoring imperfectly rejects bad customers. The implication of this imperfectness is payment collection problems. We believe it is also important to consider overdue payments since they are the end product of the imperfectness of credit scoring. Therefore, we prefer to focus on loan projection scoring since it covers not only credit scoring but also payment projections.

Solutions given from previous studies are more focused on credit scoring performance rather than solutions to payments collection problems. Solution for credit scoring performance comprises of algorithmic approaches and data centric approaches. The imperfectness of credit scoring also arises from imbalanced data problems from credit scoring data. We prefer to utilize segmentation since it can be aligned to payment problems and it is more applicable than one-class learning.

Finally, we will apply multiple metrics rather than a single metric to analyse algorithm performance from a broader point of view. Metrics that will be utilized in the research are hit rates, F-measure, AUC, and Cost sensitive learning will be applied in this dissertation as they contribute to different aspects of performance. Hit rates will be utilized to evaluate performance of predicting the positive class. The F-measure is useful to observe hit rate and precision at the same time. AUC is used to detect which algorithms fail to predict correctly by comparing them with the random guessing (AUC=0.5). Cost sensitive learning is very useful to keep the error as small as possible.

Chapter 3: Methodology

3.1. Introduction

This chapter examines the goals of the research and the research methodology required to realize these goals in practice. The research goals involve implementing solutions to both credit scoring and collection problems. The research methodology that we discuss will provide the basic framework to transit from goals to solutions.

3.2. Research Goal

There are two problems that have been identified from Literature review (see section 2.3), which are payment collection problems and credit scoring problems. Therefore, this research is focused on creating practical solutions to the payment collection and credit scoring problems. The collection problems occur because of inherent weaknesses in the credit scoring process, with no proactive action to pre-empt overdue payments. The proposed solution for the collection problem is to create a payment prediction model that will identify potential overdue payments in advance, so that a lender can take action to collect such payments earlier. By involving credit scoring parameters in building the payment prediction model, the behaviour of all such credit scoring parameters can be observed in order to give feedback to the current credit scoring function, thus improving its accuracy.

3.3. Research Methodology

A suitable methodology for building and testing a payment prediction model is design science. Firstly, according to Klabber (2006), design science seeks to construct and evaluate artefacts, which is consonant with the study objectives of constructing and assessing payment prediction models for a real world company. Secondly, in design science the emphasis is on creating effective artefacts to change reality rather than understanding reality (March & Smith, 1995).

Payment prediction models can be viewed as effective artefacts that give feedback to credit scoring systems in order to improve their performance, which goes beyond simple understanding of the behaviour of the prediction models being created.

Thus in order to achieve the stated research goals, this study will require the implementation and evaluation of a payment prediction model. In Information Systems, building a model as the solution is a part of design science (Klabbers, 2006). In 2004, Hevner et al. (2004) introduced the Information System Research Framework (ISRF) which is a conceptual framework of design science research that proposes solutions to problems. Information research in this framework is viewed as a problem solving exercise that utilizes available knowledge in order to solve problems in a given environment.

In the context of this research the ISRF framework articulates the research as a problem solving mechanism involving the implementation of prediction models to facilitate loan collection. Originally, Hevner et al. (2004) defined seven processes in their framework, which are design as an artefact, problem relevance, design evaluation, research contribution, research rigor, design as a search process, and research communication. We adapt Hevner's original ISRF to suit this piece of research, resulting in four clearly defined processes as shown in Figure 3.1. The first process involves problem identification. Problems are framed as research questions that need to be answered by conducting the research. In order to answer all questions, design as an artefact and research rigor are combined to form one process solution design as they are strongly connected. Design as an artefact refers to payment prediction design, and in order to achieve this, some domain-specific knowledge is needed. Research Rigor is the process of gathering some available knowledge from data mining to fulfil payment prediction requirements. Afterwards, the models are developed, tested, refined, and evaluated until the target is achieved in the search cycle. Then, feedback to credit scoring and payment prediction are the research contributions.

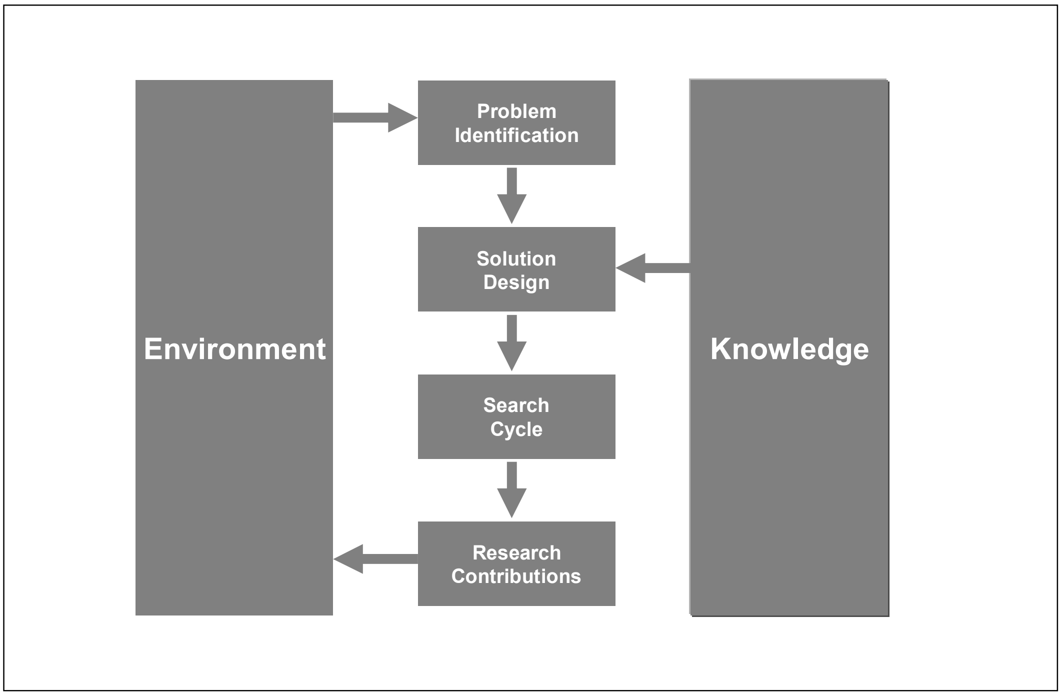


Figure 3.1: Adaptive ISRF through the research process

3.3.1. Problem Identification

Hevner et al. (2004) see the problem identification process as gaps between current condition and study goals. Parisi (2006) explains recently that most lending companies collect all accounts based on a paper aged trial balance, working from right to left on an aging report, not considering the risk of account, but simply the due balance. However, the position taken in this study is that the provision of proactive solutions through the use of payment prediction will pre-empt bad payments to a great extent. A payment prediction is needed to give information about payments that will most probably be overdue for the next payment period. Therefore, the gap that exists in current collection systems in the company under investigation is that no payment prediction model currently exists that can be used to perform proactive action in order to pre-empt bad payments. Building a payment model requires data mining algorithms. Since there are many data mining algorithms available, it is necessary to select the best algorithm for payment prediction. Hence, the first research question is:

Q1: Which data mining algorithm is the best for payment prediction?

However, the solution may not entirely depend on choosing the best algorithm, but also on the data method that is applied to increase the quality of prediction. Therefore, the second question is

Q2: Which data methods are best for payment prediction?

Furthermore, the best model for a given payment period may or may not be the best for other periods. For this purpose, the third research question is

Q3: Can the model from one payment period be re-used in subsequent payments?

Payment prediction also provides a solution to the credit scoring problem by providing feedback to the credit scoring process to improve its performance. Therefore the fourth research question is to find the answer to

Q4: Which combination of credit scoring parameters best identifies bad customers for each payment?

3.3.2. Solution Design

Learning from those previous results (see section 2.3), it is apparent that credit scoring performance will never be perfect since the classification process is itself subject to errors (Nayak & Turvey, 1997). Moreover, the best performance reached from a study today may not be optimal in the future as many new customers may come with their new behaviours. Therefore, it is necessary to design a solution with continuous improvement processes as part of the solution itself. The current credit scoring mechanism should dynamically change as data changes. Dynamic changes in credit scoring models are possible by providing feedback to the current credit scoring process. By reviewing the current credit scoring process on the basis of the feedback given, the performance of the current credit scoring process will significantly increase.

The feedback will be given when we know the quality of the current credit scoring system. The quality of the current credit scoring will be reviewed based on the accounts receivable performance. Overdue payment report is one of a batch of accounts receivable reports that show how many bad customers a lender has. The more bad customers the worse will be the performance of a credit scoring system. Therefore we will base our study on the overdue payment report issued by the lender.

Account receivable performance reduces significantly because of late payments from customers. As Parisi (2006) has discussed the collection process is done after the late payments are known from account receivable reports. This reactive action is ineffective because overdue payments have already occurred. Therefore, it is necessary to build models that support proactive actions instead of reactive actions.

Proactive action is possible if there is a prediction that identifies those customers who will not make their payment on time. Therefore this dissertation will focus on building advance payment prediction in order to pre-empt overdue payments. Payment prediction will be given for each payment period by learning from all credit scoring parameters and all available payment histories.

Data arising from a credit scoring system represents an imbalanced data mining problem as many more good customers exist than bad customers. Subsequently, good payments records are many more than bad payment records. As a result, bad payments form the minority class. As imbalanced datasets over-emphasizes the majority class, bad payment prediction is a difficult task. The proposed solution is to transform the minority class in such a way as to make it the majority class on particular segments of the data. Thus the original data will be divided into two segments. The first segment will contain more bad payment records than good payment records, while the other segment will contain the rest. By learning from segments where bad payment records are the majority, we expect prediction performance to improve.

In data mining, it is a well known fact that in general there is no single best algorithm that performs well in all situations (Witten & Frank, 2005, p. 35). Therefore, a number of different algorithms will be utilized in the payment prediction process. Appropriate metrics will be applied to test the efficacy of various different schemes.

Payment Prediction Design

The payment prediction model is built on information from customer's payments. Since information about customers is readily available in the form of credit scoring parameters, the latter are used in conjunction with payment histories to produce a payment prediction model.

Historical payment data is divided into two categories, namely good payments and bad payments. Good payments are payments that are paid in advance or within seven days of their due date, otherwise payments are categorized as being bad. Seven days is within the tolerance level for good payments since some payments may be late due to operational reasons. For example, data transfers from banks need a number of working days and some inter-branch transactions need several days to be accomplished. But delays of more than a week are due to customer failure to initiate payments on time.

Characteristics of bad payments are reflected in different combinations of credit scoring parameter values and payment history data. Since the first payment data contains no payment history, such bad payment characteristics can only be determined by a combination of credit scoring parameters. For the second payment, a combination of credit scoring and the record of actual first payments will be used as the basis of bad payment characteristics. Similarly, for the third until the last payment, both credit scoring parameters and previous payment histories will be used to learn bad payment characteristics. This process continues until the seventh payment as data is only available for seven payments only. For the seventh payment, a combination of credit scoring parameters and all payments made up to this point will be used to characterize bad payments.

Different combinations of credit scoring parameters and payment histories will be used to segment data. A data segment consists of both bad and good payments. The number of bad payments compared with good payments may be less, the same, or greater. A segment that contains more bad payments than good payment data will be called a *Majority Bad Payment Segment* (MBPS). Since a MBPS contains more bad payment data, we would expect it to be an effective vehicle in studying bad payment characteristics. This expectation is borne out by the experimental results presented in Chapter 4. In this context, it becomes important to identify which segments are indeed MBPS.

Payment Prediction Algorithms

Algorithms are an important part of payment prediction modelling. However, as has been discussed in the literature review chapter, there is no single algorithm that is universally the best across all data domains. Anticipating this issue, it is thus appropriate to involve multiple algorithms and then make comparisons amongst them to find the best performer for the data domain under study.

Galindo and Tamayo (as cited in Servigny & Renault, 2004, p. 75) specify some requirements in algorithm selection, which are accuracy (low error rates arising from assumptions) and interpretability (understanding the output of a model).

Interpretability issues are important considerations if the algorithms are to be useful in practice (Gurka, Edwards, & French, 2007).

Previous studies show algorithms such as the C4.5 decision tree algorithm (Kauderer, Nakhaeizadeh, Artiles, & Jeromin, 1999), Logistic Regression (Sohn & Kim, 2007; Xu & Wang, 2007), Neural Networks (Yang, Li, Ji, & Xu, 2001) and Bayesian Networks (Hu, 2004) have high levels of accuracy in the domain of payment prediction,

However, excluding Neural Networks, the other algorithms produce interpretable models. Logistic Regression models are interpretable as their coefficients show the changes of experiencing an event (Pampel, 2000, pp. 18-20). The same holds true for Bayesian Networks, with Santana et al. (2006) observing that: “they are one of the most prominent techniques when considering the ease of knowledge interpretation achieved”. Likewise, Cano et al. (2007) explain that decision trees are highly interpretable. However, Cano et al. warns that the degree of interpretability of decision trees depends very much on their size. Large decision trees generally exhibit the phenomenon of over fitting and hence their generalization ability will be consequently punished.

3.3.3. Search Cycle

The search cycle is the process of preparing data to be utilized to build, test, refine, and evaluate payment prediction models that refer to solution design. As shown in Figure 3.2, the search cycle consists of six processes, which are data pre-processing, data segmentation, model building, model testing, model refinement, and model analysis.

The search cycle starts from the data pre-processing step and then continues to data segmentation followed by model building and testing. Both the building and testing of payment prediction models are done in the Waikato Environment for Knowledge Analysis (WEKA) version 3.4.10 (Waikato Computer Science Department, n.d.) machine learning workbench. For the first payment prediction, only credit scoring attributes are used in the prediction process. For the second payment, all credit scoring parameters and the first payment is utilized to predict the second payment status. Similarly, for the third and subsequent payments, all credit scoring parameters and previous payment histories are utilized for prediction. The three prediction algorithms are compared on the basis of the fail prediction cost. More details on each of the steps involved in the search cycle are given below.

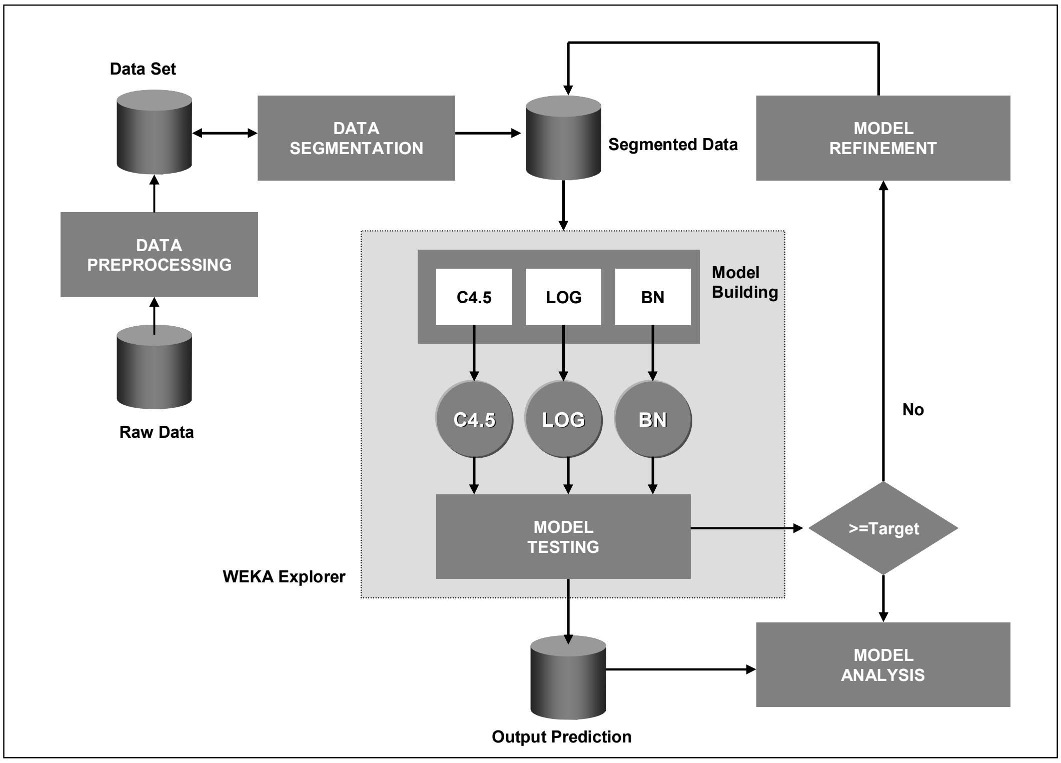


Figure 3.2: Search cycle in Payment Prediction Modelling

Data Pre-processing

The objective of data pre-processing is to combine the two tables, each containing the credit scoring parameters and the payment histories into a single table which will be used in the prediction process. All alpha numeric data from credit scoring data remains the same, whilst numeric data is transformed into alpha numeric form, consisting of three levels which are A (= Small), B (= Medium), and C (= High) or five levels which are A (= Small), B (= Small Medium), C (= Medium), D (= Medium High), and E (= High). Credit scoring parameters with a small range of numerical values (≤60) will be divided into three levels whilst other numerical credit scoring parameters with a higher range than 60 are divided into five intervals.

However, we divide age according to the Company rules: A (18 to 22 years), B (23 to 27 years), C (28 to 32 years), D (33 to 37 years), E (38 to 42 years), F (43 to 47 years), G (48 to 52 years), and H (53-57 years). Ages that are less than 17 and greater than 57 are coded as I (= others) since the youngest customers need to at least 18 years of age, and those over 57 are considered to be retired from work and thus not eligible for credit. We found that such discretization makes prediction more accurate. As has been defined previously, historical payment data is transformed into two levels of alpha numeric data, which are ***G*** representing good payment and ***B*** representing bad payment.

Data Processing (Data Segmentation)

As has been justified previously in discussion about payment prediction design, data is transformed into many segments. A segment is defined as a unique combination of credit scoring parameters and/or payment histories. Since multiple payments are to be predicted, the data segmentation process is performed for each payment. For the first payment, a segment consists of all credit scoring parameters only. For other payments, a segment consists of all credit parameters and all previous payment history data. For example, a segment for the second payment prediction consists of all credit scoring parameters and the first payment history.

Data within a segment consists of both bad and good payment records. If the number of bad payment records is greater than good payment records, then the segment is called a Majority Bad Payment Segment (MBPS). We now define the concept of size of an MBPS. An N % MBPS means that the segment consists of at least N percent of bad payment records. The size of an MBPS is measured by the percentage of its bad payment records. In carving out MBPS, we start from a size of 60% and progressively increase the size by 5% up to the maximum value of 100%. The pseudo codes for creating MBPS segments for a particular payment period and MBPS size can be found in Figure 3.3.

```
PROCEDURE CREATE_MBPS
// Suppose we have combination of N credit scoring parameters (c1, c2, c3, …cN) and M
payment histories (p1, p2, p3, …pM) in table Z sorted by c1,c2,c3…cN,p1,p2,p3…pM

INPUT   Integer Payment_Period, Real Mbps_size
//Mbps_size is the percentage of bad payment records in one segment 0.6, 0.65, 0.7, to
maximum 1
//Following matrix stores unique combination of all credit scoring and payment histories
from one to
//the last payment depends on payment period to its key
Integer Y=0
Y= FUNCTION_COUNT_ROWS_IN_TABLE_Z
 DEFINE Matrix_Key [Y]
DEFINE Matrix_Procentage [Y]
String Key1=””
Integer Matrix_Rows=0, Matrix_Iteration=0
Integer  CountBad, CountGood=0, Percentage=0
IF Y>0 THEN
    FOR each row in Z
        KEY1= Z.c1+ Z.c2+ Z.c3+ …Z.cN+ Z.p1, …Z.pPayment_Period-1
        CountBad=0
        CountGood=0
        Percentage=0
        WHILE KEY1=Z.c1+ Z.c2+ Z.c3+ …Z.cN+ Z.p1, …Z.pPayment_Period-1
            IF Z.pPayment_Period=”B” THEN
                CountBad= CountBad+1
            ELSE
                CountGood= CountGood+1
            ENDIF
            NEXT ROW
        END WHILE
        Percentage= CountBad/(CountBad+CountGood)
        IF Percentage >= Mbps_size THEN
            //only for segments that equal or larger than MBPS size
            Matrix_Rows=Matrix_Rows+1
            Matrix_Key [Matrix_Rows]=Key1
            Matrix_Procentage[Matrix_Rows]= Percentage
        END IF
    END FOR
    IF Matrix_Rows>0 THEN
        CALL PROCEDURE_CREATE_TABLE_MBPS(Payment_Period)
        FOR Matrix_Iteration=1 TO Matrix_Rows
        CALL PROCEDURE_INSERT_TABLE_MBPS(Payment_Period,
        Matrix_key[Matrix_Iteration])
        //Insert table MBPS from table Z that matched with the key
        END FOR
    END IF
END IF
END PROCEDURE
```

Figure 3.3: MBPS creation pseudo code

Model Building and Testing

As stated earlier, Payment Prediction Models are generated by using WEKA. WEKA is commonly used by data mining researchers for building models. In addition, both model building and testing can be performed in one single process in Weka Explorer module using 10 fold cross validation.

Payment prediction is merely information about the next payment in advance. Therefore payment prediction here is not for a specific number of payments but for each payment in turn. Since bad payments are the study target then the objective is to predict correct bad payments as precisely as possible. As has been described in the data pre-processing step, payment history data consists of two classes only, which are B (Bad) and G (Good). Consequently only four outcomes are possible; True Positive (TP), when bad payments are predicted correctly as bad payments (B-B), True Negative (TN), when good payments are correctly predicted as being good (G-G), False Positive (FP), when good payments are incorrectly predicted as bad payments (G-B), and False Negative (FN), when bad payments are predicted wrongly as good payments (B-G).

Research Target

Although accuracy of prediction models have been measured by the number of correct predictions, wrong predictions are also important since they will lead to negative customer perceptions. Therefore, besides correct predictions, errors should be minimized in building a model. Prediction errors are measured by using cost sensitive learning. Since True Positives and True Negatives do not constitute errors, their cost is defined as zero. In addition, to measure prediction errors, this study will use as a benchmark the credit scoring study from Egyptian Banks

(Abdou et al., 2007). In this study several models are observed by utilizing a standard prediction error cost, which is 5 for False Positives and 1 for False Negatives, adopted from German Credit Research from Hofmann (West, 2000). Their best model cost 0.23415. This study will thus use 0.23415 as the errors prediction target. Prediction error cost is calculated by using equation 2.7. By updating this equation with cost ratio above, the cost function will become the following equation:

$$Cost = 1 \times \frac{FN}{TP + TF + FP + FN} + 5 \times \frac{FP}{TP + TF + FP + FN}$$

Equation 3.1

The cost is considered as the effects of taking incorrect actions based on miss-classifications from prediction reports. The bad payment prediction reports contains all payments that are predicted correctly as bad payments (B-B) as well as good payments that are predicted incorrectly as bad payments (G-B). However, B-G is included in the fail prediction cost (cost=1) as it removes some bad payments from the reports. As noted above, both G-G and B-B cost zero as they do not represent mistakes.

Since our data is limited to seven payment periods only, we apply the cost function above as the cost of taking incorrect actions based on prediction reports and this cost does not contain amount of payments. A further study that involves a complete data of payment histories is needed to calculate more accurately the actual cost for each period. Hence, our cost ratio is applicable to all payment periods.

Some possible risks of making errors can be broadly divided into *internal* risk and *external* risk. Internal risks are the risks that affect the operational level of the lender. Prediction errors will result in inefficiency, particularly with respect to operational costs such as paper wastage, communications cost (if the lender's customer contact is through phone), postal fee (if sending letters to the customers), transportation fee (if visiting to customers' houses), and in general inefficiency in terms of human resources resulting from these activities. External risks may occur since the lender contacts the customers. Customers will, in all probability complain in the event that they are good payers but are approached incorrectly as late payers.

Model Refinement

Model will be refined based on data modification and will not include any modifications on the algorithms themselves. Models are built and tested in WEKA explorer process using 10 fold cross-validations for each algorithm. Firstly, experiments are start from the 60% MBPS level to build payment prediction models. If the models do not attain the research target, such models will be refined by increasing the percentage of majority bad customers by 5% until the maximum 100% MBPS level is reached or the target prediction error is achieved.

Model Analysis

C4.5, Logistic Regression and Bayesian Network will be utilized to build payment prediction models on MBPS from the first payment to the seventh payment. Afterwards, all models will be analysed. The objective of the model analysis is to answer the research questions.

- ***Answering Research Question One***

The first research question is to find the best algorithm amongst Logistic Regression, C4.5 and Bayesian Network on MBPS. The best algorithm is justified as the algorithm which produces the best performance on a range of suitable metrics. Several metrics are preferred rather than any one single metric. If a single metric is applied, there is a possibility that an algorithm will produce low performance on other metrics. Metrics that have been selected are bad payment hit rate, bad payment coverage, bad payment fail prediction cost, AUC and the F-measure.

Bad payment hit rate is applied to calculate the percentage of correctly predicted bad payment amongst all bad payments in MBPS. Equation two is applied to calculate the bad payment hit rate. There is no convention about the minimum target of hit rates from previous studies. However, Zekic-Susac et al (2004), refer to a bad applicant hit rate of 84% as being acceptable. Since payment prediction requires a very high bad payment hit rate, all algorithms will be expected to achieve a bad payment hit rate of at least 84%.

As has been mentioned in previous discussion, data is segmented into MBPS and non-MBPS. Since bad payment hit rate limited to bad payments on MBPS only, some bad payments that presents in non-MBPS data segments are ignored. Therefore, it is necessary to calculate the percentage of correctly predicted bad payment amongst all bad payments. *Bad payment coverage* is the extension of the bad payment hit rates that involves all bad payments both in MBPS and non-MBPS. The formula to calculate bad payment coverage is:

$$Bad\,Payment Coverage = \frac{Total\,correctly\,predictedBad\,Payments}{Total\,Bad\,Payments}$$

Equation 3.2

If the bad payment hit rate will be observed on each payment period, bad payment coverage will be analysed on the trend from the first payment to the seventh payments. The best algorithm is expected to continuously produce an upward trend from the first payment to the seventh. The upward trend will show that more and more bad payments from non-MBPS will be pulled to MBPS and at the same time most of them are predicted correctly as bad payments. If the trend displays a downward trend from the first payment to the seventh payment, then the algorithm will be excluded from the best algorithm selection.

The third metric is *fail prediction cost*. Any algorithm that fails to meet the fail prediction cost target that has been set is automatically excluded from the selection process. Equation 3.1 is applied to calculate fail prediction cost.

The next metric is the Area Under the Curve, or *AUC*. The purpose of applying AUC is commonly used a metric to assess performance with imbalanced data. Fawcett's (2005) definition of a realistic algorithm with reference to performance on AUC is applied as the minimum requirement or the algorithm will be excluded.

The last metric is F-measure. F-measure is applied to observe the affect of both hit rates and precision at the same time. This is evident from equation 2.4, where it can be observed that the F-measure is explicitly related to both hit rate and precision. It can be derived to observe the affect of false positive and false negative. From the equation 2.2., hit rate depends on false negative. If the false negative rate is large, then the hit rate will become small. From equation 2.3, precision depends on the false positive rate. If the false positive rate is large then the precision becomes small. An algorithm with a high level of F-measure will keep both the false negative and false positive rates as small as possible.

The F-measure is not redundant with respect to fail prediction cost. If prediction cost calculates the cost of false positive and false negative rates, then the F-measure is about management of failures. False positive rates may decrease from one payment period to another. This decrement may lead to an increment in the false negative rate. Conversely, the reduction of false negative may affect induce a higher false positive rate. By applying the F-measure, algorithms can be tracked on how they perform on both false positives and false negatives at the same time. If the false positive rate reduces then the precision will increase and at the same time, if false negative rate decreases then the hit rate will increase.

The first criterion is that an algorithm will produce a high performance on all metrics. Consequently, if an algorithm produces a low performance on any metric, this algorithm will be excluded. However, if all algorithms successfully produce high performance on all metrics then the best algorithm is that one that most frequently produces the best performance on all metrics applied.

More rigorously, before all algorithms are compared on a particular metric, it is necessary to apply a one way ANOVA test to calculate significant differences amongst all algorithms on that metric. A one way ANOVA test is suitable as it is a common tool to calculate significant difference among more than two groups. One way ANOVA will be run on SPSS version 14.0. (SPSS Inc, n.d.)

- ***Answering the second research question***

The second research question is to find the best data configuration method amongst MBPS, the original dataset, and under sampling. Basically this question verifies whether payment prediction models built with MBPS using the best algorithm is still better than if they were built with the original data or the application of under sampling. For this purpose, we use metrics such as bad payment coverage, fail prediction cost, and the F-measure.

Applying bad payment coverage is fairer than applying bad payment hit rate. Bad payment hit rates involves only bad payment records that present in MBPS while bad payment coverage involves all bad payment records both in MBPS and non MBPS data segments. Bad payment coverage will be more representative of the original data and under sampling since they both include all bad payment records.

Prediction failures are highly important to bad payment prediction. One single prediction failure can result in the wrong approach being made to the customer involved. Since prediction-failures will be measured on fail prediction cost and F-measure, both of these metrics are applied in comparing data configuration method performance.

AUC performances cannot be directly compared across the different data configuration methods as the underlying datasets are not the same. The original data is segmented into MBPS and non-MBPS data segments. Thus MBPS is only a subset of the original dataset.

- ***Answering Research Question Three***

Each prediction model is built for a particular payment period by the best algorithm acting on an MBPS data segment. The third research question is to observe whether the best model for one payment also performs well for subsequent payments. Firstly, the best model for the first payment will be tested by using WEKA explorer from the second payment to the seventh payment. Then the best model for the second payment period will be tested on the data from the remaining payment periods, and so on.

- ***Answering research question four***

As has been mentioned previously (see section 3.2), the combination of credit scoring and prediction models will give feedback on the current credit scoring process. The answer to the last research question depends on the answer to the third research question. If a model from a payment period can be applied to other payment periods, then the model is independent of payment history. On the other hand, if the model depends on payment period, the only model that can be used to give feedback to credit scoring parameters is the model for the first payment since the model consists of credit scoring parameters only.

3.3.4. Research Contributions

The main research contribution is a proactive solution to payment collection by generating payment predictions for the next payment period in advance. Apart from that, feedback is also providing to the credit scoring process. For each payment, the best prediction models show the combination of credit scoring parameters that leads to the majority coverage of bad customers.

3.4. Summary

The research goal is to solve both collection and credit scoring problems. Collection problems will be solved by building payment prediction models using all credit scoring parameters and/or payment histories. By including all credit scoring parameters in the payment prediction process, all credit scoring combinations on the prediction models can be utilized as feedback to give a solution for credit scoring problems.

In order to achieve that goal, we apply design science since essentially our goal comprise of constructing artefacts, which are actually payment predictions. We plan our research based on a conceptual framework of design science called ISRF to manage all our research activities on right paths to achieve the goal.

All research activities above will be focused on answering the four research questions. First question is about which algorithm is the best for payment prediction. We will answer this question by utilizing all algorithms in payment prediction construction processes. We then can compare those algorithms based on their payment prediction performance on bad payment hit rate, bad payment coverage, prediction cost, AUC, and F-measure. The second question is about finding the best data method for payment prediction. For answering this question, we will compare payment prediction performances of the best algorithm on bad payment coverage, prediction cost, and F-measure across MBPS, data original and under sampling. The third question is about investigating if one payment prediction model for a given period can be reused to another period. We will test each prediction model for one particular period on data from another period. The last question is about feedback to the current credit scoring. We will utilize combinations of all credit scoring parameters and payment history that cover bad payments on payment predictions as the feedback.

The complete discussion about answering all result questions with all payment prediction results will be given at the next chapter.

Chapter 4: Experiment Results and Discussions

4.1. Introduction

This chapter essentially consisting of four discussions in this chapter, each of which directly relates to the research questions investigated in this research. The first discussion focuses on payment prediction with MBPS. Next, the discussion moves on to prediction models. We then investigate the potential for re-using prediction models built for a given payment in subsequent payments. Finally, the findings are analysed in a bid to answer the last research question.

4.2. Majority Bad Payment Segment

In this research, we propose MBPS as an alternative solution to effectively learning a minority class in the face of overwhelming domination of instances from the majority class. For the credit dataset that is being analysed in this research, bad payments represent the minority class while good payments form the majority class. Rather than learning patterns governing bad payments from a minority class, MBPS transforms the minority class into the majority class by using the concept of a segment. A segment is a unique combination of values and the instances that they encompass taken over all credit scoring parameters and/or payment histories. All such segments that contain a majority of bad payments are regarded as Majority Bad Payment Segments (MBPS).

In this section the discussion is divided into three sub sections. The first sub section discusses the process of payment predicting by utilizing MBPS. Thereafter, a performance analysis of payment predictions built from MBPS is conducted, and finally we identify the best prediction algorithm from amongst Logistic Regression, C4.5 and the Bayesian Network.

4.2.1. Building Payment Predicting Models with MBPS

We use the original data as the basis to formulate MBPS. There are seven payment periods available in the data that we analysed. The relative proportions of good payments to bad payments changes from one period to another as can be seen in Table 4.1. It is also clear that the original data contains far more good payments than bad payments. There are a total of 7839 records, with each record representing an individual payment.

Table 4.1: Ratio of good payment records to bad payment records by payment period

Payment Periods	Good payments (G)	Bad payments (B)	Ratio G to B
1	6716	1123	6:1
2	6854	985	7:1
3	6773	1066	6:1
4	6884	955	7:1
5	7086	753	9:1
6	7248	591	12:1
7	7399	440	17:1
Average	6994	845	8:1

From the first to the fourth period the ratio of good payments to bad payments fluctuates between 6:1 and 7:1. However, from the fifth period onwards, the ratios are considerably larger, stretching from 9:1 at the fifth period to 17:1 at the seventh period. On average, the ratio between good payments to bad payments across all payments is about 8:1.

A dataset is termed *imbalanced* if it contains many more instances in one class than the other (Jo & Japkowicz, 2004). With respect to this definition, clearly the original data used in this research can be termed imbalanced. Credit card data from the International Swaps and Derivatives Association (ISDA) and the Institute of International Finance (IIF) which spans 25 commercial banks from 10 countries, also exhibit an imbalanced nature. The ratio of good credit to bad credit is almost 27:1 for small portfolios ($0-5,000) and about 42:1 for large portfolios ($0-$30,000) (Finlay 2006). In other application domains the ratio is even larger, at 100:1 or more (Chawla et al., 2004).

We start by building separate payment prediction models for each payment period. Before the data is utilized to build the prediction, it is pre-processed into several segments as outlined earlier. Segments that contain more bad payments and comprising a minimum of 60% bad payments will be regarded as Majority Bad Payment Segments (MBPS), whilst others will be considered to be non MBPS.

After such pre-processing, Logistic Regression, C4.5, and Bayesian Network algorithms will be utilized to construct the payment prediction from the MBPS models. The WEKA explorer platform was used to build and evaluate the models. Evaluation was performed using ten fold cross-validations. We use a starting value of 60% for MBPS size, which means that all segments in MBPS contain at least 60% bad payments. In addition, we track prediction errors by associating a cost factor to such errors. As has been justified previously (see discussion about research target in chapter 3), error cost target is 0.23415. If the target cost has not been achieved, then the size of the MBPS is increased in intervals of 5% to a maximum of 100% or until the target cost is achieved. A detailed version of the payment prediction results, including costs are given in **A**ppendix A.

4.2.2. MBPS Payment Prediction Performance

Payment predictions on MBPS are measured on five different metrics consisting of bad payment hit rates, bad payment coverage, error prediction cost, AUC, and bad payment F-measures. The discussion in this section is divided into five sub topics, according to these five metrics.

Bad Payment Hit Rates

Since the payment prediction objective is to predict bad payments it is important to calculate the number of bad payments that are correctly predicted. A metric to address this consideration is the bad payment hit rate. Bad payment hit rate corresponds to the number of bad payments that are predicted correctly divided by the total bad payments involved in the payment period under consideration.

For each payment, a payment prediction model is constructed on the MBPS by using Weka Explorer and validated using 10 fold cross-validations. For each payment period and for each algorithm, the bad payment hit rate is calculated from the confusion matrix obtained from the classifier output. Bad payment hit rate performance for all algorithms are shown in Figure 4.1.

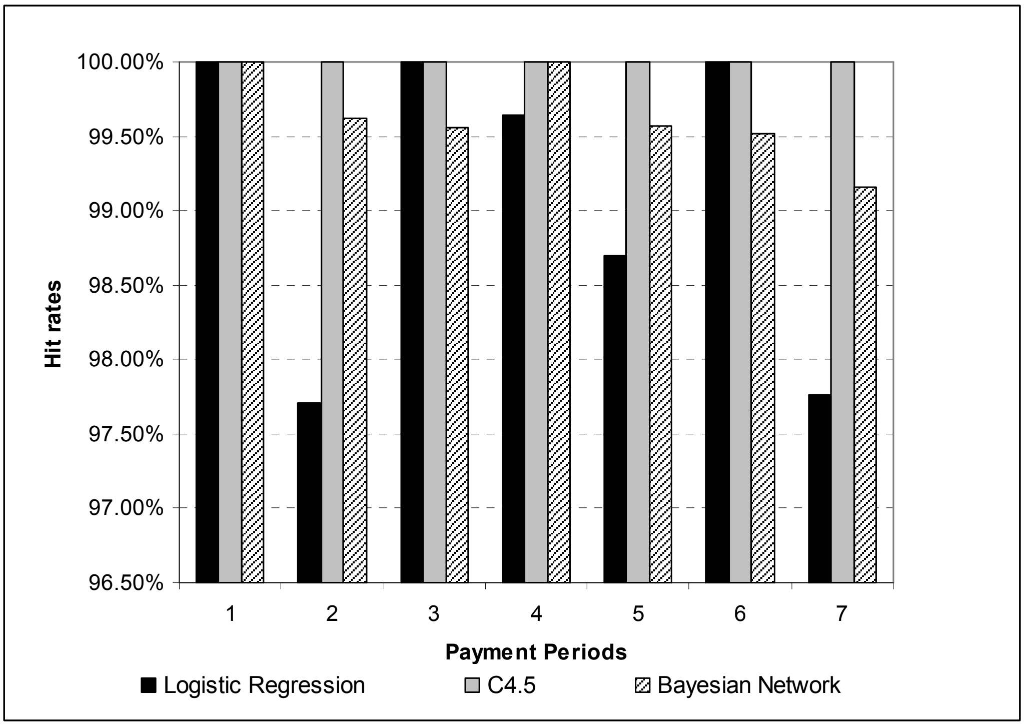


Figure 4.1: Comparison of Logistic Regression, C4.5, and Bayesian network on bad payment hit rates with MBPS

All algorithms show a very high level of performance on hit rate across all periods. The minimum hit rate is 97.71% and all algorithms are able to reach the 100% mark at various stages in the payment period. C4.5 exhibits optimal performance as it reaches a 100% hit rates across all payments. Perfect hit rates are also exhibited by Logistic Regression at the first, third and sixth period, whilst the Bayesian Network gives perfect performance at the first and fourth periods.

As has been justified previously at model analysis in chapter 3, we benchmark our study with the results from Zekic-Susac et al.'s study (Zekic-Susac et al., 2004). The performance on this metric is acceptable if an algorithm can reach hit rate of 84%. From the data above, all algorithms perform much higher than acceptable level. The minimum performance of all algorithms is found at the second period on Logistic Regression. But Logistic Regression performance at that period is 97.71%, which is 13.71% above the acceptable limit.

A comparison of algorithms is performed through a one way ANOVA test. A detailed version of these results can be found in Appendix B. By using a one way ANOVA with 95% level, it is found that there is a significant difference between Logistic Regression and C4.5 (α=0.017), but there is no such difference between any combination of Bayesian Network with either of the other two algorithms. The C4.5 algorithm outperforms Logistic Regression as its hit rate reaches 100% at all payment periods, while Logistic Regression attains a 100% hit rate in only three out of seven cases, which correspond to the first, third, and sixth payments.

Bad Payment Coverage

As a consequence of segmenting the data into the majority bad payment segments and learning exclusively from such segments it is possible that a given machine learning model may not be able to pick some bad payments. This is because such payments may be present in data segments which contain a minority of bad payments. In this context, it is important to assess bad payment coverage. For each payment, bad payments coverage is defined as total bad payment records correctly predicted divided by the total number of bad payment records present in that particular payment.

Generally, for all algorithms, bad payment coverage performance gradually improves from the first payment to the seventh payment. However, all algorithms start poorly at the first payment. All algorithms cover only 10% of bad payments in this payment period. At this period, the performance is relatively low since the prediction depends entirely on credit control parameters. As more payments are made, the bad payment coverage increases. At the second payment, payment history is used for the first time. The use of history has previously been shown to improve coverage. According to Zeng et al. (2007), by applying historical data, their collection prediction performance increases from 65.95% to 78.57%. By the fourth payment, MBPS covers more bad payments than non MBPS, as all algorithms are able to achieve a higher than 50% coverage for bad payments. At the seventh period, the coverage jumps to approximately 80% for all algorithms.

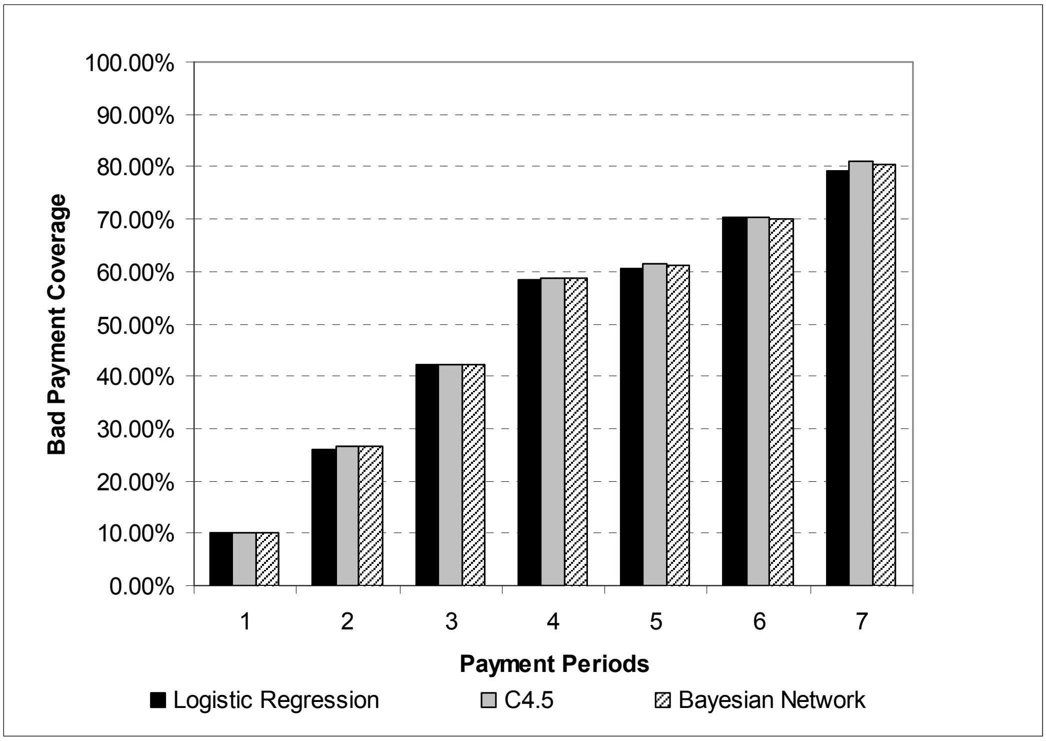


Figure 4.2: Comparison of Logistic Regression, C4.5, and Bayesian network on bad payment coverage with MBPS

However, in comparing all algorithm performances, we did not find significant differences amongst algorithms at the 95% confident level. This comparison was done by performing a one-way ANOVA test.

Fail Prediction Cost

There are four types of prediction results represented in a confusion matrix, namely True Positives, True Negatives, False Positives and False Negatives. With respect to a MBPS, true positives represent bad payments predicted correctly as bad payments (**B-B**), while true negatives represent good payments predicted correctly as good payments (**G-G**), false positives represent good payments predicted incorrectly as bad payments (**G-B**), and finally, false negatives represent bad payments predicted incorrectly as good payments (**B-G**).

Thus there are two types of misclassifications, which are False Positives and False Negatives. Misclassification cost is viewed as a fail prediction cost in this dissertation. A cost metric is used to minimize the false prediction rate to be as low as possible. As has been previously discussed in methodology chapter, we apply a ratio of False Positives to False Negatives of 5:1 for all periods. This ratio is also applied in the Egyptian Credit Scoring study (Abdou et al., 2007). The cost value of 0.23415 from their study is benchmarked as the research target to be achieved for fail prediction cost. In bad payment predictions, only B-B and G-B cells are included in bad payment prediction reports as the focus is on targeting bad customers. Based on such reports, appropriate pre-emptive action can be taken on overdue payments. Therefore, the cost of G-B is considered to be bigger than that of B-G. The ratio between these costs is taken as 5:1 in accordance with Hofmann’s suggestion. Some bad payments are miss-classified as good payments and thus these bad payments cannot be actioned, resulting in some loss of effectiveness.

The fail prediction cost is kept as low as possible with the research target as the upper limit. The lowest cost is at the first payment since all algorithms exhibit the smallest false positive rate while having zero false negatives. At the third period, all algorithms show the biggest false positive rate with 20 good payments predicted wrongly as bad payments. Unfortunately, good payments are predicted incorrectly as bad payments by all algorithms across all payments (see Appendix A). The effect of false positives is five times bigger than that of false negatives (West, 2000). In other words, the cost of a false positive is five times that of a false negative. In addition, the false negative rate is smaller than the false positive rate for all algorithms across all periods. Since the highest number of good payments present in MBPS is at its peak at the third period, the third period carries the highest cost. However, the biggest cost at third period is 0.2166 found with the Bayesian Network is still lower than the research target cost of 0.23415. This shows that all prediction models built from MBPS are low-cost in nature.

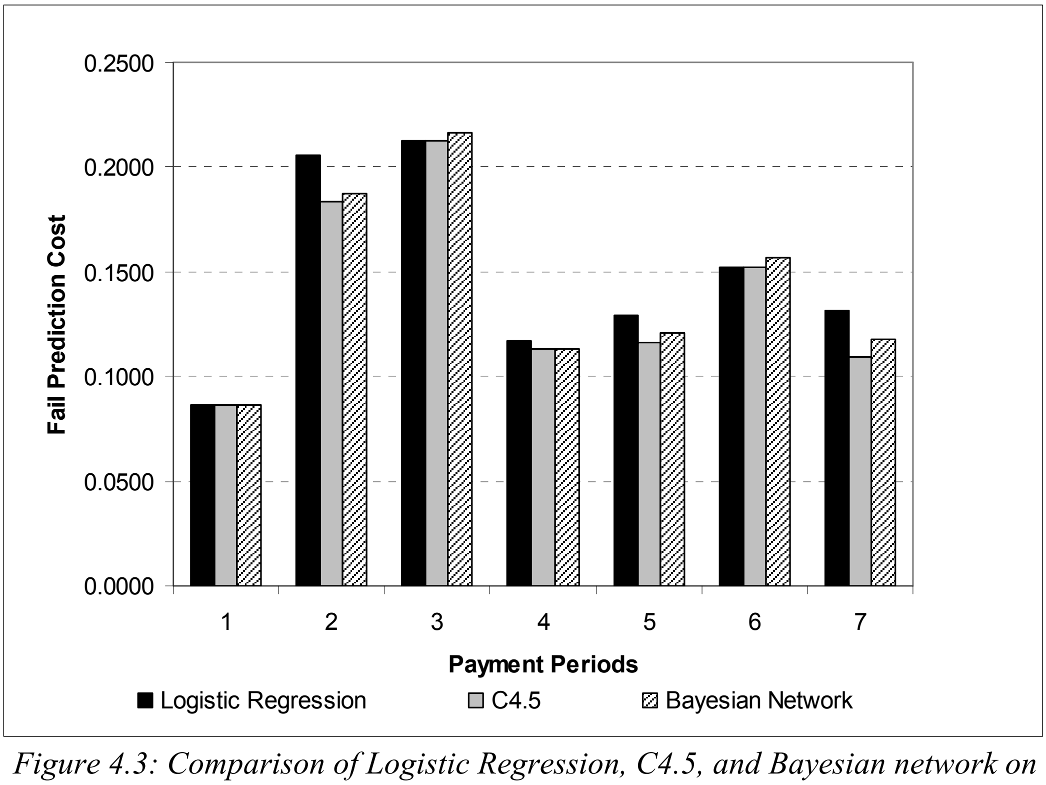


Figure 4.3: Comparison of Logistic Regression, C4.5, and Bayesian network on Fail Prediction Cost with MBPS

Overall, all models cost the same as the results from the one-way ANOVA show that there is no significant difference amongst algorithms.

Area under Curve (AUC) metric

Fawcett (Fawcett, 2005) defines an algorithm *f* to be realistic if the AUC (*f*) >0.5, otherwise *f* is worse than random guessing. Figure 4.4 shows that realistic performances on AUC are exhibited by Logistic Regression across all payments. Its minimum performance on AUC (0.6425) is shown in the seventh period, but this is still much higher than random guessing. Furthermore, except for the first payment, the Bayesian Network performance is also realistic. The worst is C4.5, as its AUC performance across all periods is worse than random guessing.

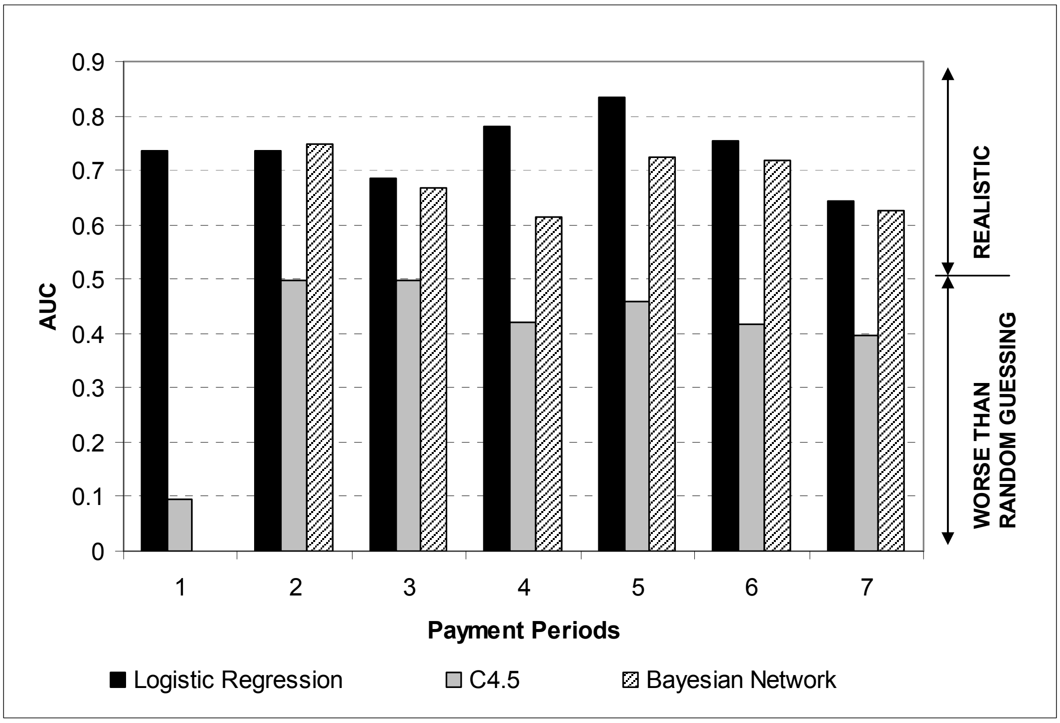


Figure 4.4: Comparison of Logistic Regression, C4.5, and Bayesian network on AUC with MBPS

Surprisingly, all AUC coefficients of C4.5 are under 0.5, which means that all bad predictions of C4.5 are worse than random guessing. This is a direct consequence of C4.5's tendency to blindly classify good payments as bad payments. Thus C4.5 does not meet Fawcett's criterion of a realistic algorithm, with respect to the AUC measure.

Bad Payment F-measure

The next discussion centers on a comparison of data re-balancing methods and uses the F-measure. The F-measure enables us to observe the simultaneous effects of hit rates and precision. Precision is defined as total number of bad payments that are correctly predicted divided by total number of bad payments. Hit rates have been discussed separately since hit rate focuses exclusively on bad payment performance, which is the focus of this dissertation. However, we decided to investigate the F-measure as it enables us to track hit rates and precision at the same time.

Figure 4.5 shows that generally, all algorithms perform well with respect to the bad payment F-measure. The minimum performance of 0.9697 is found at the second payment on Logistic Regression but this result is nevertheless very high. The maximum performance is at the first payment as all algorithms perform above 0.99. The one-way ANOVA at 95% confidence level, revealed no significant differences amongst the three algorithms. As a result, all algorithms perform similarly at very high level of bad payment F-measure.

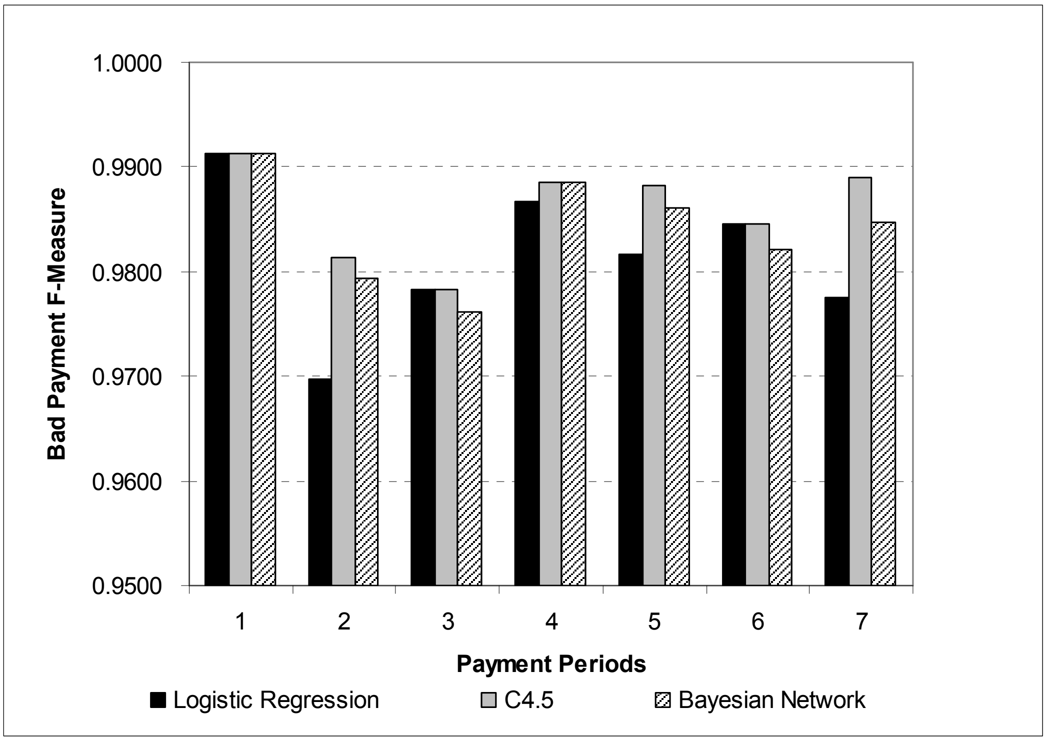


Figure 4.5: Comparison of Logistic Regression, C4.5, and Bayesian network on Bad Payment F-measure by utilizing MBPS

4.2.3. Selection of the best algorithm in predicting bad payments by utilizing MBPS

There are two criteria of selecting best algorithm in predicting bad payments with MBPS. Firstly, the performance comparison is based on bad payment hit rates, bad payment coverage, fail prediction cost, and bad payment F-measures. Secondly, it is also important to consider about minimum requirements of an algorithm to be involved in selection process. Fawcett definition of a realistic algorithm performance on AUC is applied the minimum criterion for an algorithm to qualify as the best. The best algorithm must display its consistency by producing payment predictions that are better than random guessing. If at least one of all AUC coefficients of an algorithm is found to be less than 0.5, this automatically excludes the algorithm concerned.

In comparing Logistic Regression and C4.5, Logistic Regression outperforms C4.5 on AUC, but is significantly outperformed by C4.5 on hit rate whilst on other metrics they are not significantly different. However C4.5 fails on the all important AUC measure and thus can be excluded from the selection process, which leaves us with Logistic Regression and the Bayesian Network.

Logistic Regression can be justified as being better than the Bayesian Network as it outperforms the Bayesian Network on AUC, while not being significantly different on the other metrics.

Thus, in conclusion, Logistic Regression is selected as the best algorithm in predicting bad payments with MBPS. Overall, its performances show the best from both comparing prediction performance metric and fulfilments of minimum requirements.

The next discussion is about comparing MBPS with the other methods for learning imbalanced data. As Logistic Regression has been selected as the best algorithm, only Logistic Regression is utilised in the comparison.

4.3. Comparing MBPS with other methods

In this section we test whether MBPS performs better than other methods for predicting bad payments. There are two other data configuration methods that will be compared with MBPS, which represent the unmodified dataset (hereinafter referred to as the original dataset) and under sampling of the majority class, representing good payments. Under sampling is chosen since it uses a similar way to MBPS in learning about the minority class by reducing majority class examples.

4.3.1. Bad Payments Coverage

The first metric to compare the data configuration methods are bad payment coverage. Bad payment coverage for MBPS is the number of bad payments that are predicted correctly divided by total number of bad payments in the MBPS segment. However, for both original data and under sampling, all bad payments are included in the model building process, so their bad payment coverage is actually their hit rate. Comparison on bad payment coverage across all data configuration methods is shown in Figure 4.6.

At the first period, poor performances are found not just on MBPS but also under sampling and original data. Moreover, the best performance at this period is MBPS. Amongst 1123 bad payments, 114 payments are predicted correctly with MBPS, 63 with under sampling, whilst by utilising the original dataset, only one bad payment is predicted correctly. Although MBPS shows low performance at the first period, it is still the best amongst the data configuration methods.

From a business perspective, if credit scoring is perfect, then there will be no overdue payments at the first period. Applying Logistic Regression to the original dataset at this period, we find only one bad payment implying that the credit scoring process is far from perfect. By applying under sampling, the prediction improves as 62 more payments can be predicted as being overdue. Under sampling outperforms original data since the data is imbalanced. Under sampling learns overdue payment better than with the original dataset by reducing good payment examples in its training dataset. However, MBPS has better performance than under sampling in this period. As can be seen from Appendix A, 114 bad payments are identified. However, all 114 bad payments are predicted correctly. The hit rate, as has been discussed previously, is very high, but the coverage is relatively small.

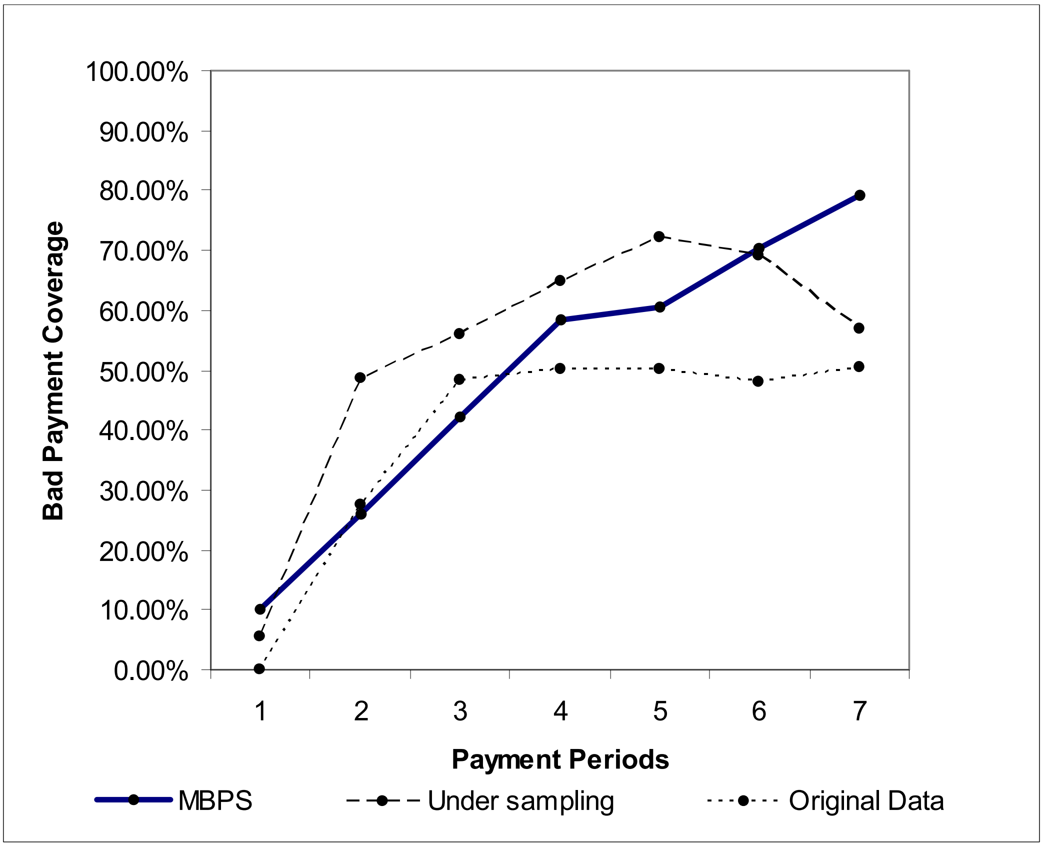


Figure 4.6: Comparison of bad payment coverage across MBPS, Under Sampling, and Original dataset

New customers are not expected to be late in their first payment, however in reality this is not the case. One factor that may cause this problem is misunderstanding for that customers may have about payment procedures. It is thus suggested that the lender reviews their customer service on payment procedures. Hopefully, this suggestion will reduce the number of overdue payments at the first period

Furthermore, from the second to the fifth payment, all data methods show considerable increment in their bad payment coverage. It is clear than under sampling outperforms the other two data methods on these periods. Its performance increases rapidly from 48.73% at the second payment to 72.11% at the fifth payment. The original dataset performance grows faster than MBPS at the second and the third payments. However, at the fourth and the fifth periods MBPS outperforms the original dataset.

In predicting bad payments, we believe that the cost of prediction errors is more important than coverage. By ensuring that prediction models have low cost, such prediction errors can be kept as small as possible. Some actions on relevant customers will be taken from the prediction results. There are risks in taking inappropriate actions and these will be discussed in the fail prediction cost section. It is better to take no action rather than take a wrong decision. In other words, we prefer precision to coverage and this can be achieved by increasing the MBPS size. Therefore, from the first to fifth payment the size of MBPS is increased in order to reach the target cost. At the second payment, for example, the original size is 60% (see Appendix A). At this size, the bad payment coverage of Logistic Regression is 350 out of 985 whilst the original dataset can only reach 272 out of 985 cases. Since the cost is more important, the size is increased to 80% and the coverage then drops to only 256 out of 985 cases.

Under sampling shows better performance from the second to the fifth period than MBPS since under sampling involves all bad payment in its learning process whilst MBPS ignores some bad payments that are present in non MBPS data segments. However, since under sampling does not take into account the prediction cost, prediction errors are abundant.

Table 4.2: Comparison of prediction results across all data configuration methods

Algorithms	Payment Periods	Prediction Results			
		G-G	B-B	G-B	B-G
MBPS	1	0	114	2	0
	2	0	256	10	6
	3	0	451	20	0
	4	0	558	13	2
	5	0	456	11	6
	6	0	415	13	0
	7	0	349	8	8
Under sampling	1	6464	63	252	1060
	2	6254	480	600	505
	3	6359	598	414	468
	4	6454	620	430	335
	5	6601	543	485	210
	6	6838	409	410	182
	7	6959	250	440	190
Original Data	1	6713	1	3	1122
	2	6659	272	195	713
	3	6474	515	299	551
	4	6529	478	355	477
	5	6841	377	245	376
	6	7048	283	200	308
	7	7242	222	157	218

As can be seen from Table 4.2, from the second to the fifth period, under sampling produces errors in predicting good payments that incorrectly as bad payment (**G-B**) in more than 400 cases. If the results from under sampling are applied then the lender will take wrong actions for more these 400-odd cases. In contrast, the same type of error is very small with MBPS. The maximum is 20 payments at the third payment period.

At the second payment, 262 out of 985 bad payments are flagged with 256 being predicted correctly. At the third payment, 451 out of 1066 bad payments are flagged, with all of them being predicted correctly. Although bad payment coverage of MBPS seems relatively small, the prediction provided by MBPS adds significant value to the lender as bad payers are flagged accurately in advance. By knowing bad payments earlier, the lender can pre-empt potential loss of revenue by taking appropriate action. For example, for 36 payment periods (spanning three years), the lender is able to find 114 potential bad payers at the first time period. Potential lost payments that will be saved by the lender in advance is 36 x 114 or 4104 payments. Similarly, at the second period, the lender will save 35 x 256 or 8960 payments, and at the third period the number is 15,334 potential lost payments.

However, poor performance on MBPS is found from the first to the third payment since information from payment history is limited. At the first payment 114 out of 1123 bad payment are involved, but all 114 bad payments are predicted correctly.
From the third payment onwards bad payment coverage with the original dataset tends to level off around the 50% mark. At the same time coverage with under sampling also tends to flatten, but this happens later, at the sixth payment. In these two data methods, bad payments represent the minority class. The dominance of good payments is very strong. As shown in Table 4.1, from the first to the fifth payment the ratio of good payments to bad payments is around 7:1 to 9:1.

However, at the sixth payment this ratio increases to 12:1 and gets even worse at the seventh payment where it rises dramatically to become 17:1. In this research we found that under sampling consistently learn well when the ratio 7:3 in the training dataset, which is agreement with Weiss and Provost (2003) who argue that the optimal natural distribution for minority classes is 30%. However, under sampling is not able to control the distribution in its testing dataset. With a ratio of 12:1 at the sixth payment and 17:1 at the seventh payment, the distribution in the testing dataset is very far from the optimal 7:3. As a result, under sampling performance is poor.

On the other hand, the domination of good payments over bad payments will never happen in MBPS since good payments are not the majority class but are in a minority in MBPS data segments to the extent that they actually dominate good payments. As a result, the more the information that is gathered from payment histories, the better is the bad payment coverage of MBPS.

Although MBPS ignores a certain number of bad payment records that present in non MBPS data segments, its performances consistently improves from the second to the seventh periods. Moreover, since the sixth period, MBPS covers more bad payments that are predicted correctly than either under sampling or original data.

Although data is limited to seven periods only, this limitation does not affect the significance of the payment prediction process. As performance constantly increases, we believe that this trend will continue into the future payment cycle. Our focus in this research is to detect customers who potentially default on their payments at the earliest possible stage as pre-emptive action can then be taken before actual bad payments manifest. As such we do not believe that the restriction of data to the first seven periods poses a significant problem.

4.3.2. Bad Payment Fail Prediction Cost

As has been explained previously during the research target discussion in the methodology chapter, the cost is considered as the effect of taking incorrect actions based on miss-classifications from prediction reports. The bad payment prediction reports contains all payments that are predicted correctly as bad payments as well as good payments that are predicted incorrectly as bad payments. The fail prediction result comparison can be seen in Figure 4.7.

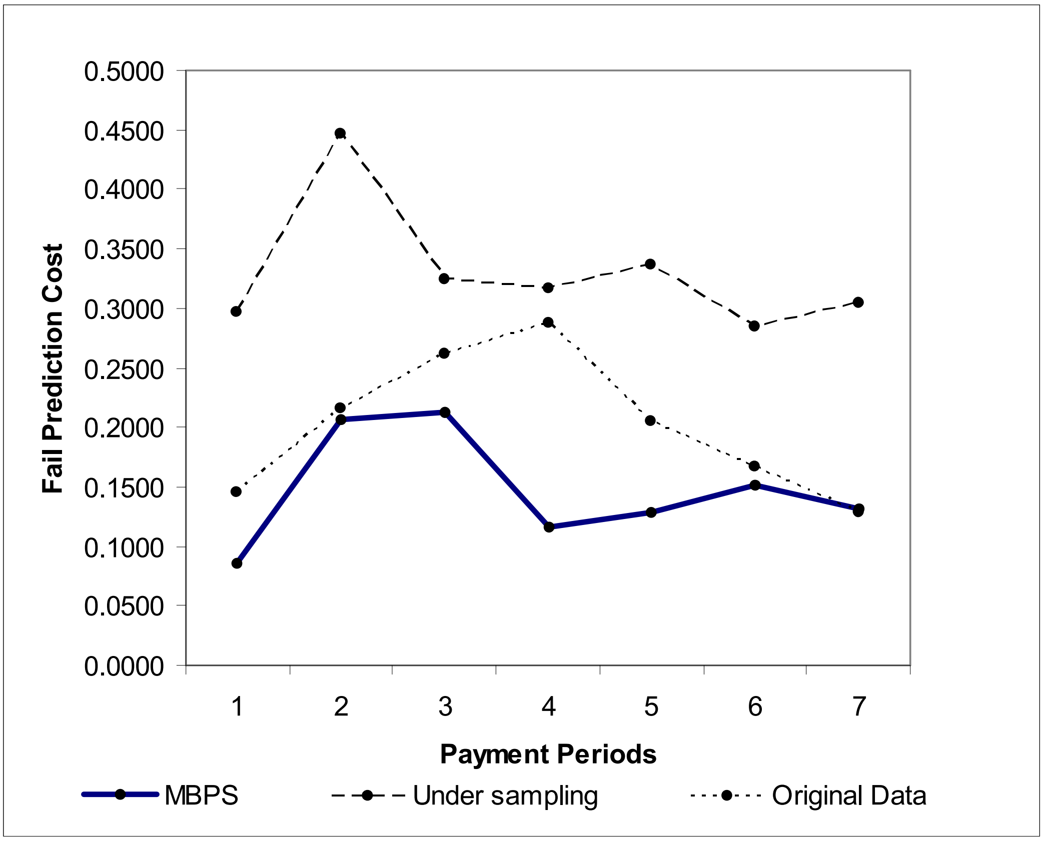


Figure 4.7: Comparison of bad payment fail prediction cost across MBPS, Under Sampling, and the original dataset

In general, payment prediction model based on MBPS produce the lowest cost when compared with other data methods. Models based on the original dataset are in second place whilst under sampling models are the worst. MBPS prediction models are the lowest cost models from the first to the sixth period. At the seventh period models based on the original dataset perform slightly better than MBPS. Back to Table 4.1, from the fifth to the seventh periods we see that the original dataset cost decreases as the ratio of good payments to bad payments becomes very high. The cost of these models decreases because more good payments are predicted correctly and consequently this results in fewer good payments being predicted as bad payments.

It is strongly suggested that the actions resulting from payment prediction should be managed carefully since Customers who appear as bad payers, have not yet been proven to be so, despite the high probability that this would materialize in the near future. It is strongly recommended that all actions are organized into a severity based hierarchy. At the lowest level of severity we have customers who are predicted as bad payers for the first time; the approach is this case should be a reminder by letter to pay at the right time. The second level represents those who appear for the second time and who have actually defaulted on the previous payment. The action is this case should be more severe. A phone call, for instance in this case may be more appropriate than a letters as this allows for more personalised contact with the customer.

In this research we have shown that under sampling performs worse than the original dataset on bad payment fail prediction cost. This finding is in agreement with previous studies, such as McCharty and Zabar (2005) who also found that under sampling performs worse in cost sensitive learning situation. The under sampling method learns by reducing the proportion of majority class instances in the training dataset. Consequently, under sampling results in the loss of some information and this causes some majority class instances to be predicted incorrectly as minority instances, thus enlarging the false negative rate. The effect of enlarging false negative errors directly enlarges misclassification errors since inherently, the false negative rate is bigger than the false positive in credit scoring scenarios (Abdou et al., 2007).

4.3.3. Bad Payment F-measure

The last discussion is centred around a comparison of data configuration methods on their F-measures. While algorithms were the focus of the previous comparison, here we examine the effect of different data configuration methods on the F-measure.

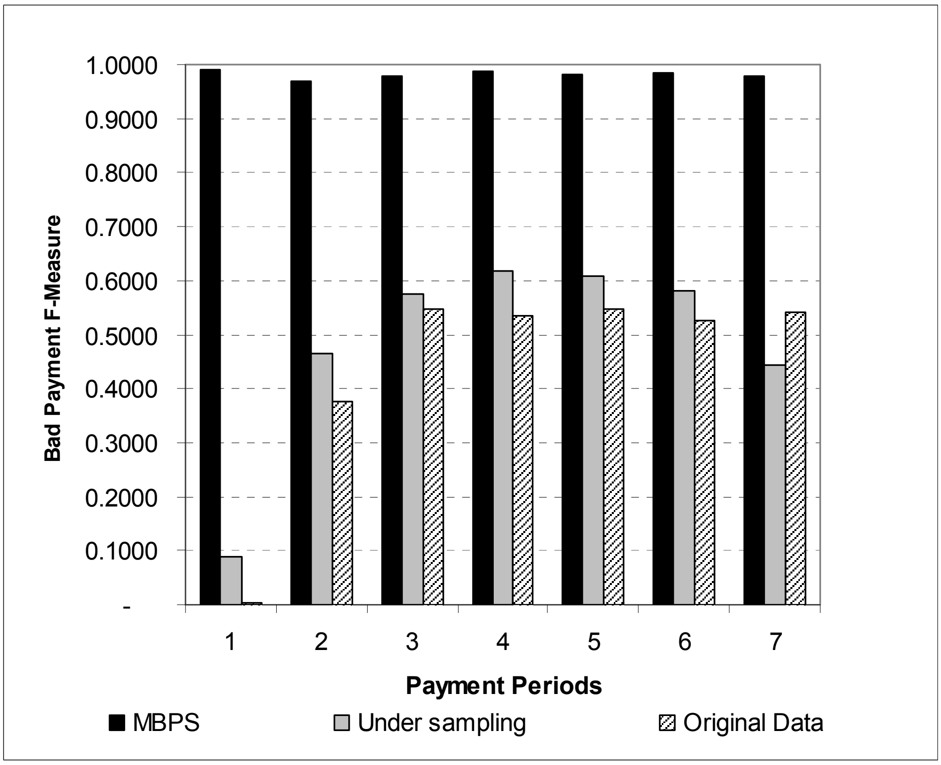


Figure 4.8: Comparison of bad payment F-measure across all data methods

Figure 4.8 indicates the comparison results on bad payment F-measure across all data configuration method at all payments. It is clear that MBPS outperforms the other two data configuration methods. MBPS shows high performance with a maximum of 0.9913 at the fist payment and a minimum of 0.9697 at the second payment. Under sampling is in second place. Except for the seventh payment, under sampling outperform original data. However, the performances of both under sampling and original data are much lower than MBPS

Comparing hit rates and precisions (see table 4.3), from the first payment to the seventh payment, MBPS hit rates are consistently higher than its precision. In contrast, original data precision is higher than its hit rate. Under sampling has a different trend, for the first and third payments its precision is higher than its hit rate. This fact shows that both the original dataset and under sampling, are not stable in their performance when compared to MBPS.

Original dataset precision is higher than under sampling precision for almost all payment periods except for the fourth payment. By using original data, the prediction errors of good payments predicted wrongly as bad payments are smaller than the corresponding errors with under sampling. Original data is better at predicting good payments than under sampling since under sampling reduces some good payment records. Both with the original dataset and under sampling, good payments form the majority class. Since the original data is better at predicting good payments, the miss-classifications of good payments into bad payments are smaller than applying under sampling. On the other hand, under sampling is better than original data in predicting bad payments since under sampling hit rate is higher than original data hit rate at all payments.

Table 4.3: Comparison of hit rates, precision, and F-measure across data configuration methods

Payment Periods	MBPS			Under Sampling			Original Data		
	Hit Rates	**Precision**	**F-measure**	**Hit Rates**	**Precision**	**F-measure**	**Hit Rates**	**Precision**	**F-measure**
1	1	0.9828	0.9913	0.0561	0.2	0.0876	0.0009	0.25	0.0018
2	0.9771	0.9624	0.9697	0.4873	0.4444	0.4649	0.2761	0.5824	0.3747
3	1	0.9575	0.9783	0.561	0.5909	0.5756	0.4831	0.6327	0.5479
4	0.9964	0.9772	0.9867	0.6492	0.5905	0.6185	0.5005	0.5738	0.5347
5	0.987	0.9764	0.9817	0.7211	0.5282	0.6098	0.5007	0.6061	0.5484
6	1	0.9696	0.9846	0.692	0.4994	0.5801	0.4788	0.5859	0.527
7	0.9776	0.9776	0.9776	0.5682	0.3623	0.4425	0.5045	0.5858	0.5421

The F-measure simultaneously tracks hit rates and precision at the same time by recording both G-B and B-G errors. Hit rate depends on B-G, whilst precision depends on G-B. With MBPS, Logistic regression produces very small B-G type. Moreover three out of seven payment periods, B-G is zero. G-B is very small as good payments are in a minority in MBPS data segments. Consequently, MBPS hit rates are very strong.

4.4. Re-use of prediction models across payments

Each model is built for a particular payment period. This section will discuss if there is a possibility of a model in particular payment being applied to another period. In order to address this consideration, a payments models trained with data from a particular period (see Appendix C for more details) is tested on data from a subsequent period in order to test goodness of fit.

Table 4.4: Cross testing results payments models across payment periods

		Payment periods						
		1	**2**	**3**	**4**	**5**	**6**	**7**
Payment models	**1**		n/a	n/a	n/a	n/a	n/a	n/a
	2	n/a		n/a	n/a	n/a	n/a	n/a
	3	n/a	n/a		n/a	n/a	n/a	n/a
	4	n/a	n/a	n/a		n/a	n/a	n/a
	5	n/a	n/a	n/a	n/a		n/a	n/a
	6	n/a	n/a	n/a	n/a	n/a		n/a
	7	n/a	n/a	n/a	n/a	n/a	n/a	

Table 4.4 shows the results of this process. Unfortunately there is no single model that is valid for a particular payment, which can be accurately applied to other payment periods.

The differences from one model to another are their payment history attributes. For the first payment, the model contains no payment history as the model is built based on all credit parameters only. For the second payment the model has a logistic regression coefficient for the first payment (=-39.6749). The third payment model has two payment history coefficients, 0.4717 for the first payment and -21.1891 for the second payment. Similarly from the fourth payment to the seventh payments, their models have previous payment history coefficients. However, the values of coefficients across different payment histories are totally different.

4.5. Recommendation for the current credit scoring

This research has focused on learning credit scoring models in a bid to improve the process of credit management. The feedback from this research can be applied to both review and improve the credit scoring system.

Some reasons given below explain the importance of reviewing the current credit scoring system. The first reason is that the total number of bad payments at the first payment period is relatively big. Around 14.33% or 1123 out of 7839 payments are overdue at this payment period. This fact shows that the current credit scoring is not strong enough to pre-empt a substantial number of bad payers. At the second period, 519 out of the 1123 default on their second payment as well, with another 466 new customers default, making a total of 985 overdue payments at the second payment period. This data shows that late payers from the first payment contribute more than 50% of overdue payments at the second period, thus highlighting the need for identifying defaulters early in the payment cycle.

About 105 combinations of all credit scoring parameters representing a total of 114 late payments (details in Appendix D) are predicted with 100% success using the combination of Logistic Regression and MBPS. If these customers are rated highly with respect to their credit scoring parameters, then it is strongly recommended that their credit rating is reduced.

MBPS is definitely applicable in a generic credit scoring system. Firstly, MBPS involves all credit scoring parameters. If the lender adds some new parameters or removes existing parameters, the MBPS is still applicable as it does not depend on specific parameters. Secondly, MBPS does not rely on any specific formula that governs the numerical values of parameters. If the lender change the values of any its credit scoring parameters, MBPS can still be applied. Thirdly, MBPS is high interpretable, as it is simple and easily understood by all categories of users.

4.6 Summary

MBPS has successfully supported bad payment prediction. MBPS has very high hit rates, good bad payment coverage which progressively increases with payment period, has low cost models, and finally has a very high F-measure which is stable across payment period.

The best algorithm is Logistic Regression as it performs well on all metrics. On AUC, both C4.5 and the Bayesian Network perform worse than random guessing, But Logistic Regression shows prediction results that are better than random guessing.

Overall, MBPS outperforms other methods on three metrics. Firstly, although it ignores a certain number of bad payment records that present in non MBPS data segments, its performances on bad payment coverage consistently improves from the second to the seventh periods. In addition, since the sixth period, MBPS covers more bad payments that are predicted correctly than other methods. Secondly, in general, a payment prediction model based on MBPS is the lowest cost model when compared with the other methods. The last metric is F-measure. The performances of other methods on F-measure are much lower than MBPS.

Payment predictions on MBPS are valid only for a particular payment period. Different periods have different payment prediction models.

Bad payment prediction can be utilised in a dynamic fashion as findings from the prediction process can be fed back to the current credit scoring process to improve the overall performance of the credit scoring system.

Chapter 5: Conclusion

This last chapter comprises of two sections. We emphasise what we have achieved in this research and highlight some of our achievements in the first section. We also describe some limitations of our study and some thoughts about future research in the limitation and future work section.

5.1. Achievements

We have presented solutions for both the credit scoring and collection problems. Our solution is initiated by generating payment predictions that allow Lenders to know earlier which payments are potentially overdue at the next period. Lender then can approach customers to pay their payment on time. Hence, Lenders can pre-empt overdue payments. A combination of credit scoring parameters was found on the first payment period that achieves a 100% hit rate on bad payments. We use such information as feedback to the current credit scoring process. By updating the current credit scoring model with the information given, it is expected that the current credit scoring performance will significantly improve.

Our solution comprises of an algorithmic and data-centric approach. We identified Logistic Regression as the best algorithm on our credit scoring data based on multiple metrics such as bad payment hit rate, bad payment coverage, fail prediction cost, AUC and the F-measure. We were also successful in overcoming problems to a great extent, due to imbalanced data with our credit scoring data. MBPS significantly alleviates the problem of domination of good payments by transforming bad payments into the majority class and good payments as the minority class. By learning from segments where bad payment records are the majority, Logistic Regression reaches a very high level of performance on all five metrics that we tracked.

We now summarise some of our key prediction results. By combining Logistic Regression and MBPS, our payment prediction performance on hit rate was never less than 97.71% and moreover, at three out of seven payment periods our hit rate reached 100%. We were also able to achieve excellent results on the F-measure, the minimum value obtained was 0.97 which was significantly better than even the corresponding maxima for under sampling and the original dataset, with values of 0.62 and 0.55 respectively. Prediction failures relating to false positives and false negatives were controlled to be as low as possible. The lowest cost is 0.0862 at the first payment, which compares very well with previous research.

Overall MBPS has better performance than with the original dataset and under sampling. Although we ignore a certain number of bad payments that present on non MBPS data segments, the coverage of bad payments are more than with other methods from the sixth payment onwards. In general, payment prediction models based on MBPS produce the lowest cost. In addition, MBPS performances on F-measure are much higher than with other methods across all payments.

A given prediction model is valid only for a particular period. Consequently, payment predictions must be built as many times as the number of payments that is available. However, payment history significantly improves Logistic Regression performance on MBPS in covering more correct overdue payments.

We emphasise our experience in conducting multiple metric analysis. We are surprised when we find that C4.5 significantly outperforms Logistic Regression on hit rates. All C4.5 prediction results show perfect performance across all payments. However, when it comes to the AUC test, the results surprise us in the other way. Here, C4.5 shows very bad performance on AUC at all payments; it is not even better than random guessing. One algorithm can reach the best performance on one metric but the result may be very different on other metrics. Having said that, Logistic Regression shows its consistency by passing all tests with a very high level of overall performance.

5.2. Limitation and Future Work

We acknowledge a limitation of our study which is the restriction of performance evaluation to the first seven monthly payments only. Data with a combination of fortnightly payments and monthly payments was out of scope for our study. Data with different types of credit scoring parameters was also out of scope. A further study is needed to observe which type of segmentation is the best for such cases.

We are optimistic that we could have a broader dimension of analysis if we have data for more than seven payment periods. The first impact of this limitation is we are unable to observe bad payment coverage from the eighth period to the end of the payment cycle. A future study with complete data that comprises of a full three years payment cycle will enable an analysis of payment predictions at the end of each year. A complete three year cycle of data will allow calculation of fail prediction cost that include potential loss from overdue amount at each payment period. This will enable the prediction cost to be calculated more accurately for each payment period.

We place the payment prediction process between credit scoring and payment collection but a bit closer to credit scoring as it is a part of a dynamic mechanism for a comprehensive credit scoring system. Future development is needed to fully integrate payment prediction with payment collection by observing the effect of payment prediction in Accounts Receivable performance.

We believe payment prediction can be utilised as a method of preserving good customers as good customers. Most customers in a lending company are predicted as good customers by credit scoring at the beginning except when manual approval applies. For various reasons, they become bad payers if they do not initiate their payment on time. However, a lender can proactively prevent them from becoming bad payers by encouraging them to pay their next payment with the help of payment prediction reports. It is expected that some of these customers will then make their payments on schedule. Hereinafter, customers will know that the lender will watch their payments carefully. Hopefully, they will organise their finances to pay to lender on time. Hence, the lender will preserve them as good payers.

We strongly recommend a usability study of customer behaviour in reacting to approaches from the lender. Our recommendation at section 4.3.2 in chapter four has not been tested as yet. By conducting a usability study, future research will be able to produce a more precise recommendation. For example, when is the best time to send a letter to customers? Is it helpful if we print our message to customer on an account statement letter? Is there any effect of sending our letter on red paper, blue paper, or green paper instead of white paper?

Finally, we hope our research will be useful for both academic and finance industrial sectors in the future.

Reference List

Abdou, H., Masry, A. E., & Pointon, J. (2007). On the Application of Credit Scoring Models in Egyptian Banks. *Banks and Bank systems, 2*(1).

Avery, R. B., Bostic, R. W., Calem, P. S., & Canner, G. B. (1996). Credit Risk, Credit Scoring, and the performance of Home Mortgages. *Federal Reserve Bulletin*(82), 621-648.

Baesens, B., Van Gestel, T., Viaene, S., Stepanova, M., Suykens, J., & Vanthienen, J. (2003). Benchmarking state-of-the-art classification algorithms for credit scoring. *Journal of the Operational Research Society, 54*(6), 627-635.

Boyes, W. J., Hoffman, D. L., & Low, S. A. (1989). An Econometric Analysis of The Bank Credit Scoring Problem. *Journal of Econometrics, 40*, 3-14.

Cano, J. R., Herrera, F., & Lozano, M. (2007). Evolutionary stratified training set selection for extracting classification rules with trade off precision-interpretability. *Data & Knowledge Engineering, 60*(1), 90-108.

Chawla, Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research, 16*, 321-357.

Chawla, Japkowicz, N., & Kotcz, A. (2004). Special issue on learning from imbalanced datasets: Editorial: special issue on learning from imbalanced data sets. *ACM SIGKDD Explorations Newsletter, 6*(1).

Crook, J. N., Edelman, D. B., & Thomas, L. C. (2006). Recent developments in consumer credit risk assessment. *European Journal of Operational Research, 183*(3), 1447-1465.

Desai, V. S., Crook, J. N., & Overstreet, G. A. (1996). A comparison of neural network and linear scoring models in credit union environment. *European Journal of Operational Research, 95*(1), 24-37.

Elkan, C. (2001). The Foundations of Cost-Sensitive Learning. *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence (IJCAI'01).* 973-978.

Fawcett, T. (2005). An introduction to ROC analysis. *Pattern Recognition Letters, 27*(8), 861-874.

Finlay, S. M. (2006). Predictive models of expenditure and over-indebtedness for assessing the affordability of new consumer credit applications. *Journal of the Operational Research Society, 57*(6), 655-669.

Gurka, M. J., Edwards, L. J., & French, L. N. (2007). Testing transformations for the linear mixed model. *Computational Statistics & Data Analysis, 51*(9), 4297-4307.

Han, H., Wang, W. Y., & Mao, B. H. (2005). Borderline-SMOTE: A New Over Sampling Method in Imbalance Data Sets Learning. *Lecture Notes in Computer Science, 3644*, 878-887.

Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design Science in Information Systems Research1. *MIS Quarterly, 28*(1), 75-105.

Hsieh, N. C. (2004). An Integrated data mining and behaviour scoring model for analyzing bank customers. *Expert Systems with Applications, 27*, 623-633.

Hu, X. (2004). A Data Mining Approach for Retailing Bank Customer Attrition Analysis *Applied Intelligence, 22*(1), 47-60.

Huang, Y.-M., Hung, C.-M., & Jiau, H. C. (2006). Evaluation of neural networks and data mining methods on a credit assessment task for class imbalance problem. *Nonlinear Analysis: Real World Applications, 7*(4), 720-747.

Isaac, F. (2006). Small Business Credit Scoring. *Business Credit, 108*(3), 20-21.

Jo, T., & Japkowicz, N. (2004). Class imbalances versus small disjuncts. *ACM SIGKDD Explorations Newsletter, 6*(1).

Johnson, A. (2006). Leveraging Credit Scoring. *Mortgage Banking, 66*(6), 76-84.

Kauderer, H., Nakhaeizadeh, G., Artiles, F., & Jeromin, H. (1999). Optimization of collection efforts in automobile financing—a KDD supported environment *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining KDD '99*.

Klabbers, J. H. G. (2006). A framework for artifact assessment and theory testing. *Simulation & Gaming, 37*(2), 155-173.

Koza, J. (1992). *Genetic programming: On the programming of computers by means of natural selection*. Cambridge: MIT Press.

Laferty. (2006). Risk Management: Effective Credit Scoring strategies. *Cards International*, 14.

Lee, T. H., & Zhang, M. (2003). Bias Correction and Statistical Test For Developing Credit Scoring Model Through Logistic Regression Approach. *International Journal of Information Technology & Decision Making, 2*(2), 299-311.

Lee, T. S., Chiu, C. C., Lu, C.-J., & Chen, I. F. (2002). Credit scoring using the hybrid neural discriminant technique. *Expert Systems with Applications, 23*(3), 245-254.

Li, F., Dou, Z.-T., Xu, J., & Huang, Y.-L. (2004). Data mining-based credit evaluation for users of credit card. *Machine Learning and Cybernetics, 2004. Proceedings of 2004 International Conference, 4*, 2586.

Li, z., Xu, j.-s., & Xu, m. (2004). ANN-GA approach of credit scoring for mobile customers. *Proceedings of the 2004 IEEE Conference on Cybernetics and Intelligent Systems, 2*, 1148.

Ma, Y., & Cukic, B. (2007). Adequate and Precise Evaluation of Quality Models in Software Engineering Studies. *Third International Workshop on Predictor Models in Software Engineering (PROMISE'07)*.

March, S. T., & Smith, G. F. (1995). Design and natural science research on information technology. *Decision Support Systems, 15*(4), 251-266.

McCarthy, K., Zabar, B., & Weiss, G. (2005). Does Cost Sensitive Learning Beat sampling for Classifying Rare Classes? *Proceedings of the 1st international workshop on Utility-based data mining UBDM '05, 69 - 77*.

Nayak, G. N., & Turvey, C. G. (1997). Credit Risk Assesment and the Opportunity Cost of Loan Misclassification. *Canadian Journal of Agricultural Economic, 45*(3), 285-299.

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.

Pampel, F. C. (2000). *Logistic Regression: A Primer*. London: Sage Publications Inc.

Parisi, J. (2006). How scoring can prioritize collection strategies and lower DSOs. *Business Credit, 108*(1), 10.

Rijsbergen, C. J. v. (1979). *Information Retrieval*. London: Butterworths.

Rimey, R. D., & Cohen, F. S. (1988). A maximum Likelihood Approach to Segmenting Range Data. *IEEE Journal of Robotics and Automation, 4*(3), 277-286.

Santana, A. L., & Frances, C. R. (2006). Strategies for improving the modeling and interpretability of Bayesian Networks. *Data & Knowledge Engineering, 63*, 91-107.

Servigny, A. D., & Renault, O. (2004). *Measuring and Managing Credit Risk*. New York: McGrawHill.

Shihab, S., Al-Nuaimy, W., Huang, Y., & Eriksen, A. (2003). A comparison of segmentation techniques for target extraction in ground penetrating radar data. *Proceedings of the 2nd International Workshop on Advanced Ground Penetrating Radar, 2003*, 95-100.

Smith, L. D., Sanchez, S. M., & Lawrence, E. C. (1996). A Comprehensive Model for Managing Credit Risk on Home Mortgage Portfolio. *Decision Sciences, 27*(2), 291-308.

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.

SPSS Inc. (n.d.). *SPSS*. Retrieved 12 June 2007, from http://www.spss.com/

Srinivasan, V., & Kim, Y. H. (1987). Credit granting a comparative analysis of classifactory procedures. *Journal of Finance, 42*, 655-683.

Sun, L., & Senoy, P. P. (2006). Using Bayesian network for bankruptcy prediction: some methodological issues. *European Journal of Operational Research, 10*, 1-16.

Thomas, L. C., Ho, J., & Scherer, W. T. (2001). Time to tell:behaviour scoring and the dynamics of consumer credit assessment. *Journal of Management Mathematics, 12*, 89-103.

Tsaih, R., Liu, Y.-J., Liu, W., & Lien, Y.-L. (2004). Credit scoring system for small business loans. *Decision Support Systems, 38*, 91-99.

Waikato Computer Science Department. (n.d.). *WEKA*. Retrieved 12 June 2007, from http://www.cs.waikato.ac.nz/~ml/weka/index.html

Wei, L., Li, J., & Chen, Z. (2007). Credit Risk Evaluation Using Support Vector Machine with Mixture of Kernel. In *Lecture Notes in Computer Science* (Vol. Volume 4488/2007, pp. 431-438). Berlin: Springer Berlin / Heidelberg.

Weiss, G. M. (2004). Mining with rarity: a unifying framework. *ACM SIGKDD Explorations Newsletter*, 7-19.

Weng, C. G., & Poon, J. (2006). A data complexity analysis on imbalanced datasets and an alternative imbalance recovering strategy. *Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence WI '06*.

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

Witten, I. H., & Frank, E. (2005). *Data Mining: Practical Machine Learning Tools and Techniques* (2 ed.). USA: Morgan Kaufmann Publisher.

Xu, X., & Wang, Y. (2007). Financial failure prediction using efficiency as a predictor. *Expert Systems with Applications 2007*.

Yang, B., Li, L. X., Ji, H., & Xu, J. (2001). An early warning system for loan risk assessment using artificial neural networks. *Knowledge-Based Systems, 14*(5-6), 303-306.

Zekic-Susac, M., Sarlija, N., & Bensic, M. (2004). Small business credit scoring: a comparison of logistic regression, neural network, and decision tree models. *26th International Conference on Information Technology Interfaces, 2004*, 265.

Zeng, S., Melville, P., Lang, C. A., Martin, I. B., & Murphy, C. (2007). Predictive modeling for collections of accounts receivable. *Proceedings of the 2007 international workshop on Domain driven data mining*, 43-48.

Appendix A

Following tables are the results of payment prediction building using MBPS for each payment. The second column is MBPS size. We start our experiment with 60% MBPS. For each size of MBPS, we build payment prediction for each algorithm. Confusion matrix (G-G, B-B, G-B, B-G) is the payment prediction results. The last column is Cost that's calculated using equation 2.7. The first period is stopped at 70%MBPS since all models cost lower than research target (≤0.23415). For the same reason, the second payment is stopped at 80%, from the third to the fifth is stopped at 70%, and from the sixth to the seventh period is stopped at 60%.

Payment Period	MBPS Size	Algorithms	G-G	B-B	G-B	B-G	Cost
1st	60%	Logistic Regression	0	116	9	12	0.4161
		C4.5	0	128	9	0	0.3285
		Bayesian Network	0	126	9	2	0.3431
	65%	Logistic Regression	0	116	9	12	0.4161
		C4.5	0	128	9	0	0.3285
		Bayesian Network	0	126	9	2	0.3431
	70%	Logistic Regression	0	114	2	0	0.0862
		C4.5	0	114	2	0	0.0862
		Bayesian Network	0	114	2	0	0.0862

To be continued on next page

Payment Period	MBPS Size	Algorithms	G-G	B-B	G-B	B-G	Cost
2nd	60%	Logistic Regression	0	350	55	3	0.6814
		C4.5	7	316	48	37	0.6789
		Bayesian Network	0	351	55	2	0.6789
	65%	Logistic Regression	0	321	36	1	0.5056
		C4.5	5	298	31	24	0.5000
		Bayesian Network	0	319	36	3	0.5112
	70%	Logistic Regression	0	289	20	1	0.3258
		C4.5	0	290	20	0	0.3226
		Bayesian Network	0	289	20	1	0.3258
	75%	Logistic Regression	0	281	17	2	0.2900
		C4.5	0	283	17	0	0.2833
		Bayesian Network	0	282	17	1	0.2867
	80%	Logistic Regression	0	256	10	6	0.2059
		C4.5	0	262	10	0	0.1838
		Bayesian Network	0	261	10	1	0.1875

To be continued on next page

Payment Period	MBPS Size	Algorithms	G-G	B-B	G-B	B-G	Cost
3rd	60%	Logistic Regression	0	538	69	0	0.5684
		C4.5	0	534	69	4	0.5750
		Bayesian Network	0	537	69	1	0.5700
	65%	Logistic Regression	0	503	46	0	0.4189
		C4.5	1	501	46	2	0.4218
		Bayesian Network	0	502	46	1	0.4208
	70%	Logistic Regression	0	451	20	0	0.2123
		C4.5	0	451	20	0	0.2123
		Bayesian Network	0	449	20	2	0.2166
4th	60%	Logistic Regression	0	615	42	0	0.3196
		C4.5	0	606	42	9	0.3333
		Bayesian Network	0	615	42	0	0.3196
	65%	Logistic Regression	0	606	36	0	0.2804
		C4.5	0	600	36	6	0.2897
		Bayesian Network	0	606	36	0	0.2804
	70%	Logistic Regression	0	558	13	2	0.1169
		C4.5	0	560	13	0	0.1134
		Bayesian Network	0	560	13	0	0.1134

To be continued on next page

Payment Period	MBPS	Algorithms	G-G	B-B	G-B	B-G	Cost
5th	60%	Logistic Regression	0	505	35	3	0.3278
		C4.5	3	488	32	20	0.3315
		Bayesian Network	0	507	35	1	0.3241
	65%	Logistic Regression	0	500	30	0	0.283
		C4.5	0	496	30	4	0.2906
		Bayesian Network	0	499	30	1	0.2849
	70%	Logistic Regression	0	456	11	6	0.129
		C4.5	0	462	11	0	0.1163
		Bayesian Network	0	460	11	2	0.1205
6th	60%	Logistic Regression	0	415	13	0	0.1519
		C4.5	0	415	13	0	0.1519
		Bayesian Network	0	413	13	2	0.1565
7th	60%	Logistic Regression	0	349	8	8	0.1315
		C4.5	0	357	8	0	0.1096
		Bayesian Network	0	354	8	3	0.1178

Appendix B

Appendix B comprises of two tables that show one way test ANOVA under SPSS version 14.0 for windows and 95% confident interval. Following table show comparison across Logistic Regression (LOG), C4.5, and Bayesian Network (BN) on bad payment hit rates, bad payment coverage, fail prediction cost, and F-measure.

Metric	Significance Values (α)		
	C4.5 vs. LOG	LOG vs. BN	BN vs. C4.5
Bad Payment Hit rates	0.017	0.143	0.288
Bad Payment Coverage	0.971	0.984	0.987
Fail Prediction Cost	0.731	0.841	0.886
F-measure	0.164	0.406	0.555

Following table indicates comparison across MBPS, Under Sampling (U/S) and Original Data (O/D) by utilizing Logistic Regression on bad payment coverage fail prediction cost, and F-measure.

Metric	Significance Values (α)		
	MBPS vs. U/S	O/D vs. MBPS	U/S vs. O/D
Bad Payment Coverage	0.75379	0.39540	0.24984
Fail Prediction Cost	0.00001	0.07831	0.00028
F-measure	0.00002	0.00001	0.61836

Appendix C

This appendix shows seven tables that comprise of all Logistic Regression Payment Prediction Models from the first payment to the seventh payment.

Table C1: Logistic Regression Payment Prediction Model for the First Payment

Attributes	Value	Coefficients	Odds Ratios
C1	D	9.459	12822.8119
C1	M	-4.2223	0.0147
C1	S	-1.3878	0.2496
C2	A	-5.6852	0.0034
C2	B	2.7212	15.1984
C2	C	-3.0684	0.0465
C2	D	6.6838	799.3192
C2	E	6.5662	710.6452
C3	A	-1.5907	0.2038
C3	B	2.5737	13.1141
C3	C	2.0534	7.7946
C3	D	4.4018	81.5954
C3	E	-3.3936	0.0336
C3	F	3.0162	20.4129
C3	G	-11.0306	0
C3	H	19.241	227120582.2

Table C1 continued

Attributes	Value	Coefficients	Odds Ratios
C3	I	26.3914	2.89503E+11
C4	A	-6.5639	0.0014
C4	B	6.8098	906.6465
C4	C	-0.1362	0.8727
C4	D	11.7236	123448.6063
C4	E	9.8469	18899.1793
C5	A	-4.6331	0.0097
C5	B	-0.8222	0.4395
C5	C	2.0154	7.5034
C5	D	6.7988	896.7977
C5	E	7.1069	1220.4059
C6	A	-10.6171	0
C6	B	10.4727	35337.9857
C6	C	2.6661	14.3844
C6	D	8.5981	5421.1069
C7	A	10.6562	42454.1894
C7	B	3.4179	30.5047
C7	C	-7.7727	0.0004
C8	A	-7.4559	0.0006
C8	B	3.2649	26.1765
C8	C	21.3014	1782646360
Intercept	[redacted]	52.6518	[redacted]

Table C2: Logistic Regression Payment Prediction Model for the Second Payments

Attributes	Value	Coefficients	Odds Ratios
C1	M	-21.0387	0
C1	D	13.2257	554450.4363
C1	S	23.2238	12189521199
C2	A	-11.5096	0
C2	B	23.533	16604825924
C2	C	-10.8146	0
C2	D	16.9063	21994815.62
C2	E	12.7467	343401.7606
C3	D	-8.3605	0.0002
C3	E	-7.6961	0.0005
C3	G	-7.4165	0.0006
C3	C	13.2951	594272.3386
C3	F	15.2233	4086873.4
C3	B	-9.0133	0.0001
C3	A	28.1984	1.76363E+12
C3	H	-0.972	0.3783
C3	I	6.8876	980.0027
C4	A	-3.3573	0.0348
C4	C	-2.8442	0.0582
C4	B	26.8787	4.71279E+11
C4	E	2.0262	7.5855
C4	D	-2.3979	0.0909

Table C2 continued

Attributes	Value	Coefficients	Odds Ratios
C5	A	-14.9912	0
C5	B	28.6829	2.86315E+12
C5	D	9.0633	8632.756
C5	C	-14.4216	0
C5	E	10.7835	48217.3192
C6	A	-12.1689	0
C6	B	9.895	19830.9035
C6	C	20.5877	873260000
C6	D	-34.9679	0
C7	C	-3.9742	0.0188
C7	A	15.259	4235357.374
C7	B	-2.7801	0.062
C8	C	67.3481	1.77374E+29
C8	B	0.1783	1.1952
C8	A	-7.1273	0.0008
S1	B	-39.6749	0
Intercept		122.8897	

Table C3: Logistic Regression Payment Prediction Model for the Third Payments

Attributes	Value	Coefficients	Odds Ratios
C1	D	17.0365	25052068.2
C1	M	-2.4611	0.0853
C1	S	-1.8076	0.1641
C2	A	-3.0949	0.0453
C2	B	-2.9517	0.0523
C2	C	-2.0062	0.1345
C2	D	15.9356	8331936.604
C2	E	18.2702	86033078.25
C3	A	31.9778	7.72267E+13
C3	B	-2.4884	0.083
C3	C	-2.6997	0.0672
C3	D	-2.6319	0.0719
C3	E	-2.6182	0.0729
C3	F	-2.7808	0.062
C3	G	-3.5344	0.0292
C3	H	15.0413	3406770.202
C3	I	21.6754	2591184385
C4	A	-2.3235	0.0979
C4	B	18.6421	124786253.3
C4	C	-1.6902	0.1845
C4	D	-2.8124	0.0601
C4	E	20.1071	539986634.9

Table C3 continued

Attributes	Value	Coefficients	Odds Ratios
C5	A	-0.7521	0.4714
C5	B	-0.8779	0.4157
C5	C	-0.6236	0.536
C5	D	0.5052	1.6574
C5	E	20.2291	610076189.2
C6	A	-2.8718	0.0566
C6	B	-1.1221	0.3256
C6	C	15.4089	4920640.369
C6	D	17.4046	36200996.47
C7	A	22.9858	9607770662
C7	B	-2.4356	0.0875
C7	C	-4.2648	0.0141
C8	A	-9.2476	0.0001
C8	B	11.9041	147867.188
C8	C	-17.2015	0
S1	B or G	0.4717	1.6027
S2	B or G	-21.1891	0
Intercept	[redacted]	50.5131	[redacted]

Table C4: Logistic Regression Payment Prediction Model for the Fourth Payments

Attributes	Value	Coefficients	Odds Ratios
C1	M	-1.8071	0.1641
C1	D	24.2449	33839594433
C1	S	-2.1896	0.112
C2	B	-8.1519	0.0003
C2	A	-8.5022	0.0002
C2	C	16.7292	18425099.6
C2	D	15.4725	5243688.516
C2	E	18.1965	79915080.72
C3	F	-1.1626	0.3127
C3	C	-0.1433	0.8665
C3	E	-0.9444	0.3889
C3	G	-0.4375	0.6456
C3	D	0.0126	1.0126
C3	A	-1.1925	0.3035
C3	B	0.944	2.5701
C3	H	26.1981	2.3861E+11
C3	I	11.8298	137279.6754
C4	A	-4.6021	0.01
C4	C	-4.0675	0.0171
C4	B	22.2587	4643328382
C4	E	2.1441	8.5347
C4	D	20.1315	553345918.2

Table C4 continued

Attributes	Value	Coefficients	Odds Ratios
C5	B	-7.2592	0.0007
C5	A	-8.6803	0.0002
C5	C	17.349	34244086.29
C5	D	18.0817	71249736.05
C5	E	18.3822	96222297.96
C6	A	-12.3984	0
C6	B	11.7545	127327.0078
C6	C	11.9459	154187.5555
C6	D	10.9075	54586.2655
C7	B	-1.1523	0.3159
C7	A	24.0584	28082341971
C7	C	-3.1501	0.0428
C7	E	35.2521	2.04076E+15
C8	C	-20.2373	0
C8	B	16.7356	18542805.19
C8	A	-13.6488	0
S1	B or G	0.8451	2.3282
S2	B or G	0.3796	1.4617
S3	B or G	-23.9741	0
Intercept	[redacted]	78.347	[redacted]

Table C5: Logistic Regression Payment Prediction Model for the Fifth Payments:

Attributes	Value	Coefficients	Odds Ratios
C1	D	11.5302	101740.4462
C1	M	-8.716	0.0002
C1	S	7.6163	2030.9725
C2	A	-6.2352	0.002
C2	B	-5.5229	0.004
C2	C	12.7759	353584.3683
C2	D	9.8075	18169.9103
C2	E	9.7342	16886.1299
C3	A	22.2115	4429353918
C3	B	-4.5465	0.0106
C3	C	-4.2239	0.0146
C3	D	-3.4195	0.0327
C3	E	-4.4855	0.0113
C3	F	-3.9398	0.0195
C3	G	12.631	305903.9945
C3	H	-0.53	0.5886
C3	I	14.5002	1983246.63
C4	A	-4.5521	0.0105
C4	B	15.1521	3806150.734
C4	C	-3.6926	0.0249
C4	D	16.6491	17006713.96
C4	E	-2.2508	0.1053

Table C5 continued

Attributes	Value	Coefficients	Odds Ratios
C5	A	-11.8438	0
C5	B	8.1506	3465.3718
C5	C	7.0028	1099.7003
C5	D	7.0334	1133.8691
C5	E	6.9725	1066.9184
C6	A	-9.9442	0
C6	B	9.0988	8944.6678
C6	C	14.8814	2903338.006
C6	D	10.5592	38530.8178
C7	A	17.4824	39131287.35
C7	B	-0.6045	0.5464
C7	C	-2.1089	0.1214
C8	A	-8.2556	0.0003
C8	B	10.8801	53106.6707
C8	C	-8.8605	0.0001
S1	B or G	0.8197	2.2697
S2	B or G	-0.0694	0.9329
S3	B or G	-0.0676	0.9347
S4	B or G	-18.5984	0
Intercept		75.5106	

Table C6: Logistic Regression Payment Prediction Model for the Sixth Payments

Attributes	Value	Coefficients	Odds Ratios
C1	D	-1.2422	0.2888
C1	M	-1.3764	0.2525
C1	S	18.4235	100280884.7
C2	A	-0.9059	0.4042
C2	B	-1.9308	0.145
C2	C	-0.4166	0.6593
C2	D	13.7352	922810.2967
C2	E	18.6842	130145232.4
C3	A	-3.7524	0.0235
C3	B	-2.5405	0.0788
C3	C	18.2851	87321877.73
C3	D	-2.9487	0.0524
C3	E	-2.9222	0.0538
C3	F	-2.9372	0.053
C3	G	-2.3782	0.0927
C3	H	-18.5079	0
C3	I	16.3247	12294822.58
C4	A	-2.6492	0.0707
C4	B	13.1904	535204.5031
C4	C	-1.8761	0.1532
C4	D	12.3153	223091.6599
C4	E	-18.4788	0

Table C6 continued

Attributes	Value	Coefficients	Odds Ratios
C5	A	-6.3933	0.0017
C5	B	-4.6638	0.0094
C5	C	15.2889	4363832.89
C5	D	12.9556	423211.6829
C5	E	17.2672	31554629.02
C6	A	-10.6786	0
C6	B	11.1735	71215.5985
C6	C	-8.6136	0.0002
C6	D	8.0851	3245.6704
C7	A	10.3521	31322.5047
C7	B	10.8615	52127.6945
C7	C	-11.4018	0
C8	A	-6.6814	0.0013
C8	B	8.5672	5256.5626
C8	C	-7.5016	0.0006
S1	B or G	-0.4252	0.6536
S2	B or G	-0.4993	0.6069
S3	B or G	-0.1093	0.8965
S4	B or G	-1.3249	0.2658
S5	B or G	20.7955	1074936934
Intercept		46.8334	

Table C7: Logistic Regression Payment Prediction Model for the Seventh Payments

Attributes	Value	Coefficients	Odds Ratios
C1	D	24.3535	37720262574
C1	M	-6.2473	0.0019
C1	S	4.0131	55.32
C2	A	-7.2812	0.0007
C2	B	-5.9112	0.0027
C2	C	15.4845	5306819.515
C2	D	20.2535	625177104.2
C2	E	12.7884	358054.3987
C3	A	9.9643	21253.476
C3	B	19.3605	255942060.6
C3	C	-8.0814	0.0003
C3	D	-8.2606	0.0003
C3	E	-9.5426	0.0001
C3	F	-7.8816	0.0004
C3	G	21.3587	1887891176
C3	H	16.8771	21361361.62
C3	I	12.5407	279494.3144
C4	A	-4.4452	0.0117
C4	B	17.6171	44770914.08
C4	C	-4.1097	0.0164
C4	D	17.1468	27973950
C4	E	-2.5367	0.0791

Table C7 continued

Attributes	Value	Coefficients	Odds Ratios
C5	A	-7.2743	0.0007
C5	B	-6.3798	0.0017
C5	C	20.0286	499230275.5
C5	D	18.3667	94747358.4
C5	E	10.6786	43417.539
C6	A	-0.1862	0.8301
C6	B	0.0441	1.0451
C6	C	8.7651	6406.8423
C7	A	18.198	80033795.77
C7	B	-0.3973	0.6722
C7	C	-1.7356	0.1763
C8	A	-14.7135	0
C8	B	16.7169	18199612.88
C8	C	-11.5231	0
S1	B or G	-0.0337	0.9668
S2	B or G	0.8124	2.2533
S3	B or G	0.2686	1.3082
S4	B or G	-0.1188	0.888
S5	B or G	22.331	4991317055
S6	B or G	25.4269	1.10343E+11
Intercept		51.6284	

Appendix D

Following table indicates this research feedback to the current credit scoring. There is 105 segments in Majority Bad Payment Segments with size=70%. Total Number of bad payments=114.

Segment No.	C1	C2	C3	C4	C5	C6	C7	C8	Total Bad Payments
1	D	A	D	A	B	A	C	A	1
2	D	A	E	C	B	A	A	A	1
3	D	A	E	C	D	A	C	A	1
4	D	A	F	A	B	A	C	B	2
5	D	A	F	A	C	A	B	A	1
6	D	A	G	A	A	B	C	A	1
7	D	A	G	A	D	A	C	A	1
8	D	A	H	A	A	A	C	A	1
9	D	B	D	A	B	B	C	A	1
10	M	A	B	A	B	B	C	A	1
11	M	A	B	D	C	A	C	A	1
12	M	A	C	A	B	A	B	B	1
13	M	A	C	B	B	B	C	A	1
14	M	A	D	A	B	B	B	A	1
15	M	A	E	A	A	A	C	C	1
16	M	A	E	A	B	A	A	B	1
17	M	A	E	C	A	B	C	A	1
18	M	A	E	C	D	A	C	A	2
19	M	A	E	D	B	A	B	A	1
20	M	A	F	A	A	A	B	B	1

Segment 21 to 50 can be seen on the next page.

Segment No.	C1	C2	C3	C4	C5	C6	C7	C8	Total Bad Payments
21	M	A	F	B	B	B	C	A	1
22	M	A	F	C	B	A	B	A	1
23	M	A	F	C	C	A	C	A	1
24	M	A	G	A	A	A	C	C	1
25	M	A	G	A	A	B	B	B	1
26	M	A	G	A	C	A	C	C	1
27	M	A	G	A	C	B	C	A	1
28	M	A	G	A	C	C	C	A	1
29	M	A	G	C	A	A	C	A	3
30	M	A	G	C	D	B	C	A	1
31	M	A	H	A	A	A	C	B	1
32	M	A	H	C	A	A	C	A	1
33	M	A	I	A	A	A	C	A	1
34	M	B	A	A	A	A	A	A	1
35	M	B	A	C	A	A	B	B	1
36	M	B	B	C	C	C	C	A	1
37	M	B	C	A	C	C	C	A	1
38	M	B	C	C	C	A	A	A	1
39	M	B	C	C	E	A	C	A	1
40	M	B	C	E	A	A	C	A	1
41	M	B	D	A	A	D	C	A	1
42	M	B	D	C	B	B	B	A	2
43	M	B	E	C	A	A	A	A	1
44	M	B	E	D	C	A	A	A	1
45	M	B	F	A	A	B	B	A	1
46	M	B	F	A	B	A	C	B	1
47	M	B	F	B	B	B	A	A	1
48	M	B	G	A	A	A	B	B	1
49	M	B	G	A	B	A	A	A	1
50	M	B	G	A	B	C	B	A	1

Segment 51 to 80 can be seen on the next page.

Segment No.	C1	C2	C3	C4	C5	C6	C7	C8	Total Bad Payments
51	M	B	G	A	E	B	B	B	1
52	M	B	G	C	D	A	C	A	1
53	M	B	G	D	A	A	B	A	1
54	M	B	H	A	A	A	A	A	1
55	M	B	H	A	C	C	C	A	1
56	M	C	A	D	A	A	A	A	1
57	M	C	B	A	B	B	C	A	1
58	M	C	C	A	A	B	C	A	1
59	M	C	D	A	E	A	C	A	1
60	M	C	D	C	D	A	C	A	1
61	M	C	E	C	D	A	C	A	2
62	M	C	F	A	A	A	A	A	1
63	M	C	F	A	C	B	C	A	1
64	M	C	F	B	B	B	C	A	1
65	M	C	F	C	C	A	C	A	1
66	M	C	F	D	E	B	C	A	1
67	M	C	F	E	A	B	C	A	1
68	M	C	G	A	B	A	C	A	3
69	M	C	G	A	B	C	B	A	1
70	M	C	G	A	C	C	C	A	1
71	M	C	G	C	B	B	C	A	1
72	M	D	B	A	C	A	C	A	1
73	M	D	E	B	A	A	C	A	1
74	M	D	E	E	A	B	C	A	1
75	M	D	F	A	C	B	C	A	2
76	M	D	F	A	D	B	C	A	1
77	M	D	F	C	A	C	C	A	1
78	M	D	F	C	B	B	C	A	1
79	M	D	F	C	B	C	C	A	1
80	M	D	F	D	A	B	C	A	1

Segment 81 to 105 can be seen on the next page.

Segment No.	C1	C2	C3	C4	C5	C6	C7	C8	Total Bad Payments
81	M	D	G	A	A	B	C	A	1
82	M	D	G	A	E	B	B	A	1
83	M	D	G	C	B	C	C	A	1
84	M	D	G	D	A	A	C	A	1
85	M	D	H	A	A	B	C	A	1
86	M	E	C	C	C	A	C	A	1
87	M	E	D	D	B	B	C	A	1
88	M	E	E	A	B	A	C	A	1
89	M	E	E	A	B	A	C	B	1
90	M	E	E	C	C	B	C	A	1
91	M	E	F	A	A	B	B	A	1
92	M	E	F	A	B	B	C	A	1
93	M	E	F	D	B	B	C	A	1
94	M	E	G	D	A	A	B	A	1
95	S	A	A	C	A	B	C	A	1
96	S	A	B	A	B	A	B	A	1
97	S	A	B	B	A	A	C	A	1
98	S	A	C	C	E	A	C	A	1
99	S	A	F	A	D	A	C	A	1
100	S	B	B	C	B	A	A	A	1
101	S	B	C	C	B	B	C	A	1
102	S	B	C	C	C	A	A	A	1
103	S	B	I	C	B	A	B	A	1
104	S	C	B	D	A	A	A	A	1
105	S	E	A	C	B	A	C	A	1