621-Final Project

## Team Members

* Valerie Briot
* Michael D’acampora
* Keith Folsom
* Brian Kreis
* Sharon Morris

# Abstract

Financial institutions seek to minimize risk by identifying credit card holders who are likely to default on a payment. This research endeavors to apply and explore multiple classification techniques against the UCI credit card default dataset to determine which model provides the most predictive accuracy in identifying defaulters. We evaluate the following classifiers from both parametric and nonparametric models – binary logistic regression, Ridge and Lasso logistic regression, decision tree, Naive Bayes, and neural net. All models will be assessed against a specific performance metrics to select a single classifier with the best predictive power for credit card default prediction.

# Keywords

classification, UCI credit card dataset, binary logistic, ridge regression, LASSO, Naive Bayes, neural net

# Introduction

Our project team has obtained a data set containing 30,000 observations with each observation representing a credit card customer. Variables included in the data set provide information on that customer’s payment history, outstanding balances as well as demographic information. It is our task to develop a set of classification models to predict the probability that a customer will be in default of their next scheduled payment. The UCI credit card data set contains our default payment response variable, which is binary (0,1), an ID field, and 23 predictor variables. We will break the data set into a training and test set for our evaluation.

The objective is to build multiple classification models on the training data to predict the probability that a person will be in default of payment. We will then run analyses to determine which model performed best and apply subsampling techniques to attempt to further improve the model’s accuracy.

This model may be useful for financial institutions who provide revolving credit facilities to determine which customers may need intervention, including where reduction in the size of the outstanding credit facility would be prudent.

The paper is organized into the following sections:

* **Literature Review** reviews similar research published on the topic of financial risk management using data mining and classification to identify credit card default and the related field of fraud detection
* **Methodology** describes the approach and techniques used in this research
* **Experimentation and Results** provides an exploratory overview of the dataset and details the results of the models
* **Discussion and Conclusions** concludes research and identifies further areas of work

# Literature review

Credit card default prediction is described as an application of classification techniques within data mining. Financial and lending institutions employ probability of default (PD) models to calculate expected loss associated with default, and more generally identify individuals who are more likely to default on a payment. Similarly, fraud detection is another component within risk management employed to mitigate loss. Prediction of both credit card default and fraud see the application of similar binary classification algorithms. Due to the similarity between the two applications of classification, research dealing with both will be explored to more fully understand the current state and emerging state-of-the-art techniques. Also, the techniques highlighted address the challenges created by class imbalance in the outcome variable where the majority of cases (non-default or legitimate transactions) can significantly outnumber minority cases.

Universal to all literature in this area is the cited challenge with the lack of real-world data. Most available data is simulated or anonymized due to legal restrictions. Consequently, research development and the pool of available literature are somewhat limited or constrained by the limitations of the datasets. However, a key point is that the ability to apply accurate models within these risk management contexts can represent a huge potential savings for financial institutions.

Pasha, Fatima, Dogar, & Shahzad (2017) in their research explore the predictive accuracy of six algorithms for default prediction - linear discriminant analysis, Naïve Bayes, C4.5 decision tree, Logistic Regression, Neural Networks -MLP, and k-Nearest Neighbor (KNN). Their work evaluates performance using metrics such as accuracy (correct classification, incorrect classification, precision, and recall. The results show that the relatively newer algorithm of Multilayer Perceptron within the class of neural networks proves to be the best algorithm with an accuracy rate of 81.7%. It is worth noting that logistic regression is a close second in performance at 81%. The use of MLP and KNN algorithms are the focus of Koklu and Sabanci’s research (Koklu & Sabanci, 2016) on estimation of credit card customer’s payment status using classification within data mining techniques. Specifically their research uses Multilayer Perceptron (MLP) and k-Nearest Neighbor (KNN) algorithms using the open-source WEKA data mining platform. The performance of these algorithms is evaluated in the context of accuracy, MAE (Mean Absolute Error), and RMSE (Root Mean Squared Error). Similarly results were found where the MLP algorithm outperformed the KNN.

Altabrawee (2016) examines fourteen classification models and their predictive accuracy in default prediction. He notes the importance of a successful model’s ability to avoid underfitting or overfitting training data. In particular, the appropriate amount of regularization in the model should be used to limit under or overfitting. Performance results are evaluated in terms of average accuracy, precision, recall, and F score. The partial decision tree algorithm PART is determined to be the most accurate while Naïve Bayes-related algorithms are at the bottom in performance.

Butaru et al. (2015) research the application of machine learning techniques to the problem of credit risk. Their focus is on decision tree, random forest, and logistic regression models. Logistic regression models are noted to be the more traditionally used models for assessing credit risk. Regularization is used in their logistic regression model in order to be more in line with the anticipated performance of the decision trees and random forest models. Performance is evaluated using precision, recall, F score, and the kappa statistic. Their research determines that, although all three models performed well, decision trees and random forest outperform logistic regression especially within short time horizons.

Within the broader context of risk modeling and applications of classification techniques, Lusis (2017) compare machine learning techniques for credit card fraud detection, analyzes fraud detection classification approaches using Logistic Regression (LR) and Random Forest (RF) algorithms. Their findings are compared to results from previous research conducted using SVM or Support Vector Machine algorithms. Citing the challenge of lacking real word data, Lusis’ analysis is constrained to simulated data for legal reasons. Both the LR and RF models are tested using PCA (Principal Component Analysis) and without PCA, which is a technique used to reduce the dimensionality of data. Of these models, Random Forest without PCA and K=3 had the best predictive performance as determined by accuracy, sensitivity, and specificity. They observe that applying a Preprocessing step helped the performance of the LR models whereas Preprocessing was not necessary for the Random Forest model.

Seeja et al. (2014) propose a novel approach to handling the class imbalance problem by using a frequent itemset mining approach based on an Apriori algorithm. The authors, also citing challenges with real world dataset availability, compare their itemset mining approach to SVM, K-nearest neighbor (KNN), Naive Bayes (NB), and Random Forest algorithms. Evaluation of each model’s performance is done using Matthews correlation coefficient and BCR (and Balanced classification rate). Seeja et al. (2014) itemset approach outperformed all other test models as measured by sensitivity, false alarm rate, balanced classification rate, and Mathews correlation coefficient. It worth noting that the authors attempted to use SMOTE to address the class imbalance within the data and ultimately saw this lead to performance degradation so it was abandoned. Padmaja et al. (2007) approach fraud detection as an unbalanced classification problem and research addressing class imbalance by using hybrid sampling techniques. They approach their research applying a “combination of random under-sampling and over-sampling using SMOTE.” SMOTE: Synthetic Minority Over-sampling Technique. Classifiers used are k-NN, Radial Basis Function networks, C4.5 and Naive Bayes.

Zareapoor and Shamsolmoali (2015) in their research also note that the lack of real world data for researchers has limited the amount of published literature available on the topic. They approach their research using five classification techniques - (1) SVM, Naive Bayes (NB), KNN, and Bagging ensemble. Their premise is that ensemble learning techniques are superior to the other techniques tested. To assess model performance, Zareapooret et al. (2015) also use the metrics of Fraud Catching Rate, False Alarm Rate, Balanced Classification Rate, and Matthews correlation coefficient. They discount using accuracy and error rate, citing these as biased metrics. Their analysis shows that a bagged ensemble classifier using decision trees outperforms KNN, NB, and SVM.

Sahin and Duman (2011) focus on Neural Networks and Logistic Regression (binomial and multinomial) classification models. They too raise the challenge created by class imbalance and recommend the use of under and oversampling techniques. Specifically, Sahin and Duman (2011) employ a stratified sampling “to under sample the normal records so that the models have chance to learn the characteristics of both the normal and the fraudulent records’ profile” (p. 5). Their research shows a clear performance advantage of Neural Net models over LR models and cites the overfitting behavior of logistic regression. Among the logistic regression models, stepwise MLR is the champion in both accuracy performance and catching fraudulent transactions except the last case.

The review above of available research shows many relevant models, techniques, and insights useful for our own research. While regression modeling is our main focus, we feel it important to consider techniques beyond logistic regression to more fully understand strengths and limitations of traditional techniques within the context of emerging state-of-the-art techniques. Clearly seen in the research though is that higher predictive accuracy is being driven through the use of ensemble machine learning techniques. While important to understand, ensemble techniques are outside the scope of this research.

# Methodology

This research will explore several classification techniques against the UCI credit card default dataset to determine which model provides the most predictive accuracy in identifying payment default. Our work includes analyses using logistic regression models as this has traditionally been a common technique applied to default detection. However, we also apply more current approaches to better understand logistic regression within the context of emerging state-of-the-art machine learning classifiers.

#### Dataset Description

The UCI default credit card clients dataset consists of monthly payment data of credit cardholders of an undisclosed bank in Taiwan starting in September of 2005 and ending in April 2005. The data contains one binary response variable which indicates whether a payment default will occur in the next month. There are 23 explanatory variables included in the dataset. The dataset contains 30,000 total observations, of which 6,636 observations (28%) are for cardholders with a payment default indicator.

**Table 1** Credit Card Default Dataset

|  |  |  |
| --- | --- | --- |
| Variable | Definition | Type |
| LIMIT\_BAL | Amount of given credit in NT dollars | Predictor |
| SEX | Gender (1=male, 2=female) | Predictor |
| EDUCATION | (1=graduate school, 2=university, 3=high school, 4=others, 5=unknown, 6=unknown) | Predictor |
| MARRIAGE | Marital status (1=married, 2=single, 3=others) | Predictor |
| AGE | Age in Years | Predictor |
| PAY\_0 | Repayment status in September, 2005 | Predictor |
| PAY\_2 | Repayment status in August, 2005 | Predictor |
| PAY\_3 | Repayment status in July, 2005 | Predictor |
| PAY\_4 | Repayment status in June, 2005 | Predictor |
| PAY\_5 | Repayment status in May, 2005 | Predictor |
| PAY\_6 | Repayment status in April, 2005 | Predictor |
| BILL\_AMT1 | Amount of bill statement in September, 2005 | Predictor |
| BILL\_AMT2 | Amount of bill statement in August, 2005 | Predictor |
| BILL\_AMT3 | Amount of bill statement in July, 2005 | Predictor |
| BILL\_AMT4 | Amount of bill statement in June, 2005 | Predictor |
| BILL\_AMT5 | Amount of bill statement in May, 2005 | Predictor |
| BILL\_AMT6 | Amount of bill statement in April, 2005 | Predictor |
| PAY\_AMT1 | Amount of previous payment in September, 2005 | Predictor |
| PAY\_AMT2 | Amount of previous payment in August, 2005 | Predictor |
| PAY\_AMT3 | Amount of previous payment in July, 2005 | Predictor |
| PAY\_AMT4 | Amount of previous payment in June, 2005 | Predictor |
| PAY\_AMT5 | Amount of previous payment in May, 2005 | Predictor |
| PAY\_AMT6 | Amount of previous payment in April, 2005 | Predictor |
| DEFAULT | Default payment (1=yes, 0=no) | Response |

Exploratory data analysis is performed on the dataset using of descriptive analysis, correlation analysis, visualization, and Principle Component Analysis. The EDA phase is important in understanding the characteristics of the data for model building. Characteristics such as zero/missing values, skew, outliers, and potential data issues all inform the subsequent variable recoding and transformation applied to the data. Correlation and Principal Component Analysis (PCA) are used to identify existing collinearity among the predictor variables, notably the BILL\_AMT variables. This is further confirmed through variance inflation factor (VIF) analysis.

#### Models Explored

1. Binary Logistic Regression
2. Logistic Regression using Ridge and Lasso Regularization
3. Decision Trees
4. Naive Bayes
5. Neural Net

We implement model fitting and testing using a 70% split of the transformed credit card default dataset. Both the training and test datasets are created through random sampling. The training dataset is used to fit all models evaluated in this research and, similarly, the test dataset is used to determine predictive performance of the same models based on a hold-out dataset.

#### Performance Measurement

A critical feature of the credit card default prediction models being evaluated is their ability to accurately predict whether a cardholder will default on next month’s payment. As such, accuracy will be one measure used to determine a model's performance. However, accuracy alone does not provide a full picture for model assessment so the following performance measures will be used:

**F1-score**

The harmonic mean of the precision and recall

**Kappa statistic**

Kappa takes into account the accuracy that would be generated purely by chance.Kappa takes on values from -1 to +1, with a value of 0 meaning there is no agreement between the actual and classified classes. A value of 1 indicates perfect concordance of the model prediction and the actual classes and a value of 0 indicates total disagreement between prediction and the actual.

**Sensitivity**

Sometimes called the true positive rate, sensitivity is rate of that the event of interest, in this case default, is correctly predicted within the sample (Kuhn et al. 2013).

**Specificity**

As the converse of sensitivity, specificity is true negative rate and is the rate at which the nonevent is correctly identified as the nonevent within the sample (Kuhn et al. 2013)

**Balanced Accuracy**

Defined as (Sensitivity + Specificity)/2, balanced accuracy provides a potentially more accurate metric of accuracy in binary classification models with potentially imbalanced class distributions.

**False-Positive Rate**

Sometimes called the false alarm rate, false-positive rate is defined as 1 - Specificity.

**Youden's Index**

Youden’s index evaluates the ability of a classifier to avoid misclassifications. This index puts equal weights on a classifier’s performance on both the positive and negative cases.

Model selection is performed using the performance metrics noted above as the selection criteria. Each model's performance will be assessed using the same test dataset.

# Experimentation and Results

**Exploratory Data Analysis (EDA)**

A detailed exploratory data analysis can be found in the appendices. For the sake of brevity, the following are items of note regarding the data set:

**Missing Values**

The data does not contain missing values and as such no imputation will be necessary.

**Variable: SEX**

The majority of customers are female. This variable was made into a dummy variable.

**Variable: EDUCATION**

The majority of customers went to university, there are very few in the other/unknown categories as well as an unknown 0 value and we will combine these values into dummy variables of College and Advanced Degree, with a 0 value in the College variable representing High School and all other possibilities.

**Variable: MARRIAGE**

The majority of customers are single and the proportion of default payments appears to be higher for married individual. It appears that there are 0 values here which were not planned. We recoded this as a binary variable.

**Variable: AGE**

The distribution is right skewed. We can see that extremely young customers seem to have a higher proportion of defaults.

**Variable: PAY**

The vast majority of clients are on time or ahead of payments. The number of extremely late payments in the latter months is more infrequent. We can presume that this list only contains customers whose accounts have not yet been charged off, for which we would expect no future payments. Customers with extremely late repayment statuses 6 months ago have likely already been charged off. It is somewhat surprising that PAY\_0, PAY\_2 and PAY\_3 contain lower frequencies of extremely late payment statuses.

**Variable: PAY\_AMT**

The majority of payments are small with some rather large outliers.

**Variable: DEFAULT**

As we would expect the majority of customers are not in default of their next payment.

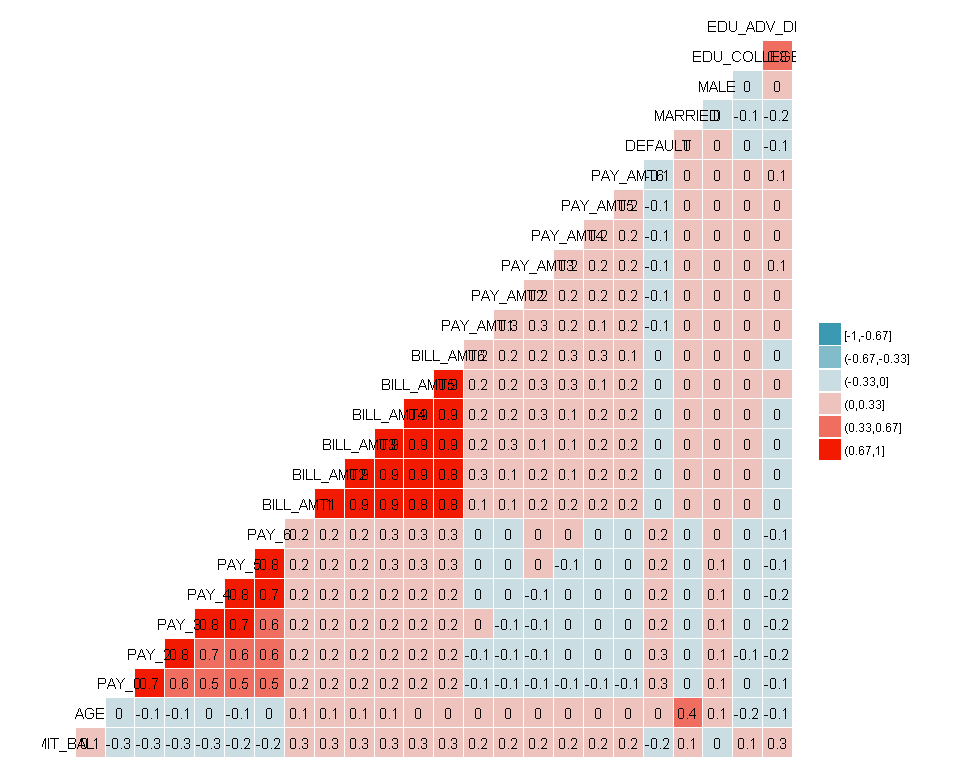
**Correlation Analysis**

As shown below there is high collinearity among variables where we would expect it to occur. For instance in Bill Amount, for which we have six variables all corresponding to different months, we would expect a relation between the amount someone owes one month and the amount that they owe the next. This is also true of the Payment Amount variable, for which we also have 6 sequential months of data for.

Also as we would expect, the DEFAULT variable is most strongly correlated with the payment status. If a person were to default in one payment period, intuitively we would think that may have a relationship with other periods.

The tables below represent correlation between response and predictor variables.

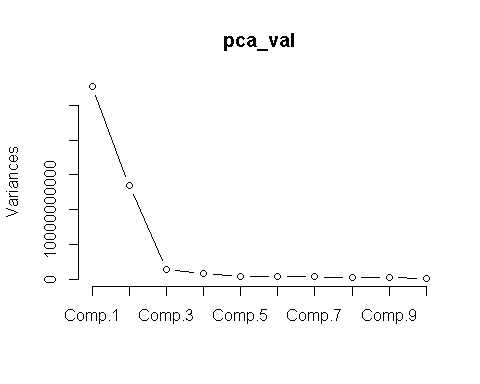
**Fig. 1** Correlation among predictor variables



**PCA component analysis**

As mentioned earlier we have quite a bit of collinearity among predictors, as demonstrated by the screeplot below; the variability accounted for by each component drops sharply within the first few components.

**Fig. 2** PCA screeplot



### Data Manipulation

The variables MARRIAGE, SEX and EDUCATION were all converted to dummy variables. As discussed previously, we combined values 4, 5 and 6 in education to create an all-encompassing other category. We will also change the unknown value of 0 in MARRIAGE to 3(other).

We have very high variance inflation factor (VIF) values for all of the BILL\_AMT variables. VIF values indicate the severity of collinearity among variables. This is not all that surprising as we would not expect the bills to change significantly from month to month for an individual in general. We created additional variables in attempt to account for the variation that these variables represent and remove the original variables. The following variables were created:

* AVG\_BILL: The average bill over the six month period for each customer
* AVG\_BILL\_TO\_LIMIT: The AVG\_BILL variable divided by that individual customer’s credit limit
* PAY\_TO\_BILL: Average Payment made over the 6 months divided by the AVG\_BILL
* INC\_COUNT: It may also be worthwhile to see for how many months the customer’s bill increased from one month to the next. This shows how often a customer is spending more than they payoff each month. If this occurs for five consecutive months it may indicate a worsening financial condition.

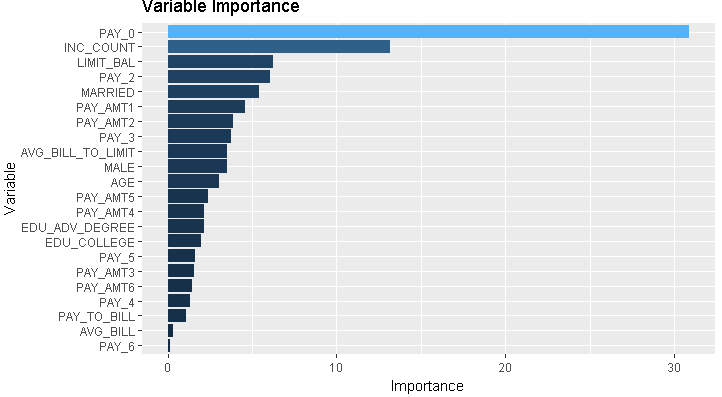
### Model Design

In this section, we compare the five default prediction models – binary logistic regression, regularized logistic regression, decision tree, Naïve Bayes, and Neural Networks. Each model is fit against the same training dataset as described in the methodology.

### Binary Logistic Regression

We use binary logistic regression from among the family generalized linear models as the baseline model for this research. Stepwise variable selection using both the forward and backward options is used to obtain the optimal model based on the lowest AIC (Akaike Information Criterion)

**Fig. 3** Variable importance among predictors



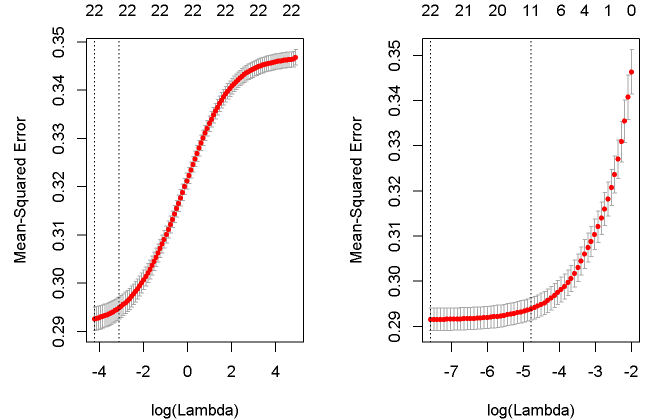
### Ridge and Lasso Regularization

Regularization is applied using the same parameters established in the binary logistic regression model but attempts to minimize the impact of the coefficients. Both Ridge and Lasso approaches attempt to reduce overfitting by applying a penalty to the parameter estimates. This is a tunable parameter know as λ.

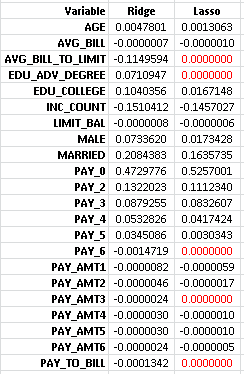
Ridge regression applies standardization to the predictor variables and uses L2 Regularization by applying a penalty term where the parameter estimates are only allowed to increase if there is a proportional decrease in the sum of squared errors (Kuhn et al 2013). Ridge regression has benefits when multicollinearity exists among the predictors and overall seeks to reduce model complexity.

Lasso, or least absolute shrinkage and selection operator, is similar to Ridge and applies a similar penalty but uses L1 regularization. Lasso has the added benefit of feature selection which ridge does not.

**Fig. 4** Ridge – Optimal lambda value **Fig. 5** Lasso– Optimal lambda value



**Table 2** Coefficient comparison between Ridge and Lasso models:



We see that the feature selection of Lasso removed PAY\_6, PAY\_AMT3, EDU\_ADV\_DEGREE, AVG\_BILL\_TO\_LIMIT, and PAY\_TO\_BILL (highlighted in red above) as predictors in the model.

## Regression Tree

A basic regression tree was used on this data set to predict default, which partitions the data into smaller groups that are more homogenous with respect to the response. The data were split into training and test sets at a 70:30 ratio.

### grow tree

m5 <- rpart(target ~ ., data = train\_all, method = 'class')  
m5

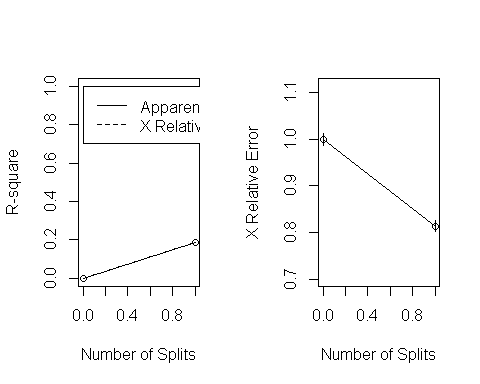
## n= 20998   
##   
## node), split, n, loss, yval, (yprob)  
## \* denotes terminal node  
##   
## 1) root 20998 4687 0 (0.7767883 0.2232117)   
## 2) PAY\_0< 1.5 18789 3146 0 (0.8325616 0.1674384) \*  
## 3) PAY\_0>=1.5 2209 668 1 (0.3023993 0.6976007) \*

#### Plot and summarize

# additional plots  
par(mfrow=c(1,2))   
rsq.rpart(m5)

##   
## Classification tree:  
## rpart(formula = target ~ ., data = train\_all, method = "class")  
##   
## Variables actually used in tree construction:  
## [1] PAY\_0  
##   
## Root node error: 4687/20998 = 0.22321  
##   
## n= 20998   
##   
## CP nsplit rel error xerror xstd  
## 1 0.18626 0 1.00000 1.00000 0.012874  
## 2 0.01000 1 0.81374 0.81374 0.011920

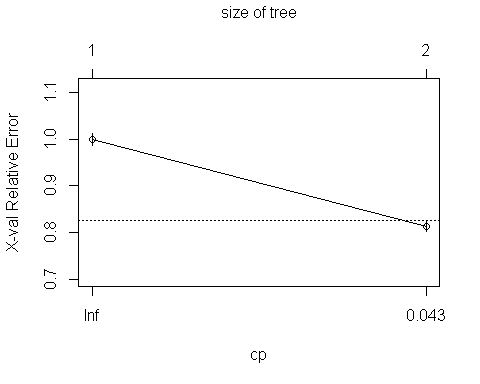
## Warning in rsq.rpart(m5): may not be applicable for this method



The relative error is , similar to linear regression. The xerror is related to the [PRESS](https://en.wikipedia.org/wiki/PRESS_statistic) statistic. The split from 0 to 1 appears to have the largest improvement of fit. The split from 1 to 2 is much less of an improvement.

The figure to the left shows a visual of the aforementioned first split providing more information than the second split. The figure to the right shows if a tree should be pruned, and in our case it does not.

plotcp(m5) # visualize cross-validation results



Looking at the plotcp function we can take a look and see if the tree needs pruning, and we can see 3 nodes appears to be the ideal size.

summary(m5, cp = 0.1) # detailed summary of splits

## Call:  
## rpart(formula = target ~ ., data = train\_all, method = "class")  
## n= 20998   
##   
## CP nsplit rel error xerror xstd  
## 1 0.1862599 0 1.0000000 1.0000000 0.01287371  
## 2 0.0100000 1 0.8137401 0.8137401 0.01191979  
##   
## Variable importance  
## PAY\_0 PAY\_5 PAY\_4 PAY\_6 PAY\_3 PAY\_2   
## 85 4 4 3 3 2   
##   
## Node number 1: 20998 observations, complexity param=0.1862599  
## predicted class=0 expected loss=0.2232117 P(node) =1  
## class counts: 16311 4687  
## probabilities: 0.777 0.223   
## left son=2 (18789 obs) right son=3 (2209 obs)  
## Primary splits:  
## PAY\_0 < 1.5 to the left, improve=1111.1410, (0 missing)  
## PAY\_2 < 1.5 to the left, improve= 831.8165, (0 missing)  
## PAY\_3 < 1.5 to the left, improve= 632.3639, (0 missing)  
## PAY\_4 < 1.5 to the left, improve= 559.3184, (0 missing)  
## PAY\_5 < 1 to the left, improve= 508.5227, (0 missing)  
## Surrogate splits:  
## PAY\_5 < 2.5 to the left, agree=0.900, adj=0.048, (0 split)  
## PAY\_4 < 2.5 to the left, agree=0.900, adj=0.045, (0 split)  
## PAY\_6 < 2.5 to the left, agree=0.898, adj=0.032, (0 split)  
## PAY\_3 < 2.5 to the left, agree=0.898, adj=0.031, (0 split)  
## PAY\_2 < 2.5 to the left, agree=0.898, adj=0.027, (0 split)  
##   
## Node number 2: 18789 observations  
## predicted class=0 expected loss=0.1674384 P(node) =0.8947995  
## class counts: 15643 3146  
## probabilities: 0.833 0.167   
##   
## Node number 3: 2209 observations  
## predicted class=1 expected loss=0.3023993 P(node) =0.1052005  
## class counts: 668 1541  
## probabilities: 0.302 0.698

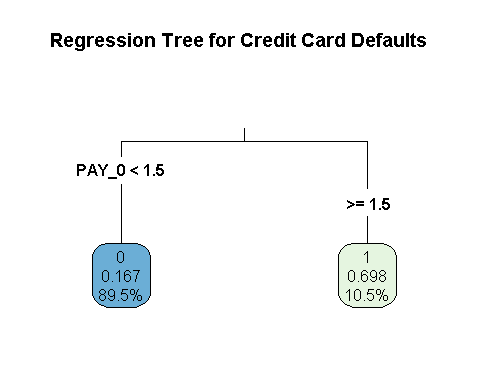
#print(m5)

Turning to the summary, the first split partitions the 20998 observations into groups of 18789 and 2209 (nodes 2 and 3) with mean values of 0.167 and 0.698, respectively. Variables PAY\_0 and PAY\_2 are weighted highest in importance.

# Mike - this plot throws an error "need finite ylim"  
  
  
  
#plot(predict(m1), resid(m1))  
#temp <- m1$frame[m1$frame$var == '<leaf>',]  
#axis(3, at = temp$yval, as.character(row.names(temp)))  
#mtext('leaf number', side = 3, line = 3)  
#abline(h = 0, lty = 2)

A residual plot of predicted values vs. residuals shows

rpart.plot(m5, type=3, digits=3, fallen.leaves=TRUE, main='Regression Tree for Credit Card Defaults')



Lastly in the plotting section is a visual of the regression tree. The model tells us that if , there is a 69.8% chance the customer will default. Approx 10.5% of the data set falls under this threshold. On the other hand if , the model takes a look at the customers $`PAY\_-2$ scores. If there is a 41.6% chance the customer will default, which comprises of 7.5% of the dataset. Lastly if there is only a 14.5% chacne of default, which 81.9% of the data set falls under.

### Predict

We run prediction on the test data, find the mean absolute error from original data set to predictions and obtain an of 0.139.

m5\_metrics <- calc\_metrics\_3("Model5 - DT", m5, test\_all, train\_all, "decision.tree")  
  
all\_model\_metrics <- rbind(all\_model\_metrics, m5\_metrics[[1]])  
  
kable(m5\_metrics[[1]])

# Neural Network

A Neural Netowrk model was used to fit the credit card dataset. A neural network processes the data on several levels.The first layer of the neural network receives the predictor variables which are processed and passed to the hidden layer. The hidden layer passes the processed predictor values to the last layer where the output is produced

A neural network model was chosen to predict default based on its ability to learn form the information provided.

## Data Preprocessing

As neural networks use activation functions between -1 and +1 - the variables were scaled down. This is done to prevent the neural network from spend training iterations doing the scaling.

Min-max normalziation was used to transform the data into a common range.This removes the scaling effect from all the variables.

The normalzied data returns a matrix that was converted to a dataframe so that the neural network model can be computed.

## target ~ LIMIT\_BAL + AGE + PAY\_0 + PAY\_2 + PAY\_3 + PAY\_4 + PAY\_5 +   
## PAY\_6 + PAY\_AMT1 + PAY\_AMT2 + PAY\_AMT3 + PAY\_AMT4 + PAY\_AMT5 +   
## PAY\_AMT6 + MARRIED + MALE + EDU\_COLLEGE + EDU\_ADV\_DEGREE +   
## AVG\_BILL + AVG\_BILL\_TO\_LIMIT + PAY\_TO\_BILL + INC\_COUNT

## Train the Neural Network

The neural network was calculated using the neuralnet library. There are 2 hidden units in the third layer and 4 hidden units in the second layer.

## Plot Neural Network

The neural network plot represents the weights of each connection. The visualization show the 3 hidden layers.

The black lines of the model represent the connections with weights.The weights are calculated using the back propagation algorithm. The blue line displays the bias term

## Nueral Network Matrix

The matrix shows the training process required 13,017 steps until all absolute partial derivatives of the error function were smaller than 0.01, which is the default threshold. The estimated weights ranged between -1.38 and 0.75

## Generalized Weights

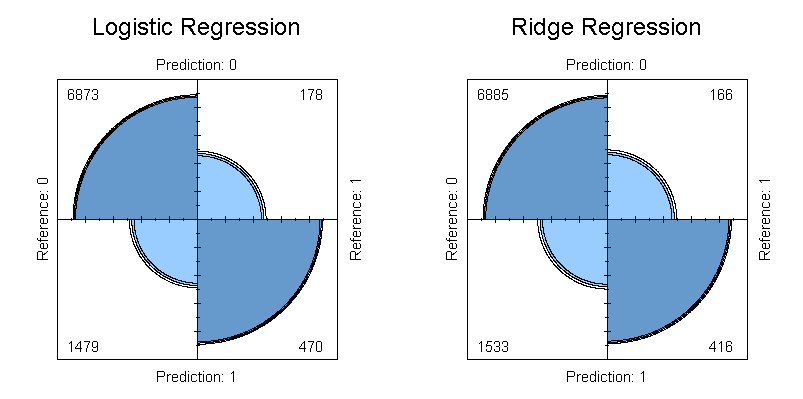
The gerneralized weights for the demographic covariates shows that they all have and effect on the target varaible.

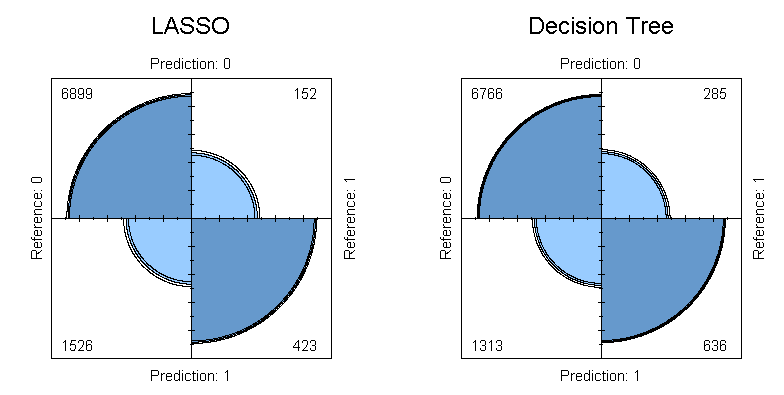
## Model Results

The five models were each tested against a test or hold-out set of data containing 9,000 observations from the credit card default dataset. The following performance metrics are captured for evaluation – (1) Accuracy, (2) F1-Score, (3) Kappa, (4) Sensitivity, (5)Specificity, (6) Balanced Accuracy = (Sensitivity + Specificity)/2, (6) False-Positive Rate = (1 – Specificity), and (7) Youden’s J Index.

The confusion matrices for the five models against the test dataset with prediction results are shown below.

**Fig. 2** Confusion Matrices for the five models using the test dataset





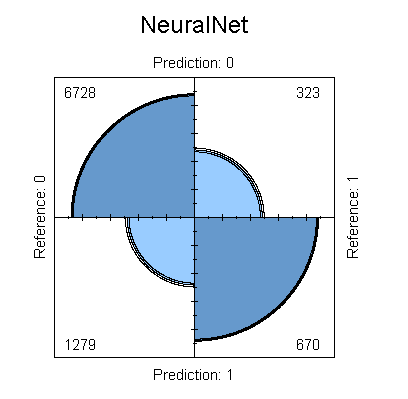


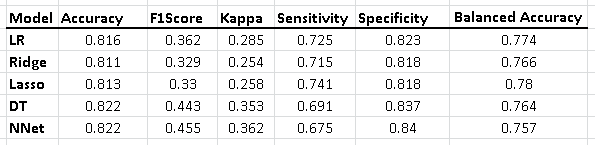
Figure 3 shows the consolidated performance metrics of the five models. Initial review shows that the parametric regression models – baseline logistic regression, Ridge, and Lasso – all have accuracy rates less than those of the decision trees and Neural Net models.

The performance of the three parametric models is fairly equivalent, looking across accuracy, F1-score, and sensitivity and specificity. Sensitivity, which indicates the model’s ability to predict default, ranges from 71.5% to 74% which fairly good. Specificity for these models ranges from 81% to 82% which is better than the no-information rate of 78% if one predicted a non-default value for each observation in the test set.

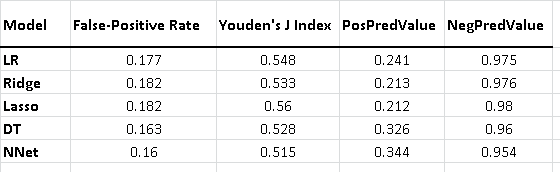
It is interesting to note that the Lasso model has the highest balanced accuracy among all models at 78%.

The non-parametric decision tree and neural net models outperform the parametric models in accuracy, F1-score, and Kappa. Their balanced accuracy is roughly on par with those of the logistic regression and ridge models. The non-parametric models do differ in sensitivity and specificity. These models seem slightly less sensitive to predicting defaults while showing a higher specificity for non-default predictions.

**Fig. 3** Model Performance Metrics



Another consideration which should be factored into the final model selection is the false-positive rate. A bank looking to implement a default prediction model would likely seek to minimize the number of false alarms generated by the algorithm. The Neural Net model has the lowest false-positive rate as well as the lowest Youden’s Index.

**Fig. 4** Model Performance Metrics Continued

The decision tree and Neural Net models are very comparable in performance across all metrics. Therefore, the lower false alarm rate of the Neural Net model makes this the selected model.

# Conclusion

# References

<https://datascienceplus.com/fitting-neural-network-in-r/> <https://www.youtube.com/watch?v=LTg-qP9iGFY> <https://www.analyticsvidhya.com/blog/2017/09/creating-visualizing-neural-network-in-r/>

<https://stats.stackexchange.com/questions/21717/how-to-train-and-validate-a-neural-network-model-in-r> <http://blog.revolutionanalytics.com/2016/03/com_class_eval_metrics_r.html>

Altabrawee, H. (2016). Performance Evaluation of Fourteen Machine Learning Algorithms on Credit Card Default Classification. Education College Journal, 463-473

Butaru, F., Chen, Q., Clark, B., Das, S., Lo, A., & Siddique, A. (2015). Risk and Risk Management in the Credit Card Industry. <doi:10.3386/w21305>

Koklu, M., & Sabanci, K. (2016). Estimation of Credit Card Customers Payment Status by Using kNN and MLP. International Journal of Intelligent Systems and Applications in Engineering, 249-251

Kuhn, M., & Johnson, K. (2013). *Applied predictive modeling.* New York: Springer.

Lusis. (2017, April 20). A Comparison of Machine Learning Techniques for Credit Card Fraud Detection. Retrieved from <http://www.lusispayments.com/uploads/4/4/8/2/44826195/a_comparison_of_machine_learning_techniques_for_credit_card_fraud_detection.pdf>

Padmaja, T. M., Dhulipalla, N., Krishna, P. R., Bapi, R. S., & Laha, A. (2007). An Unbalanced Data Classification Model Using Hybrid Sampling Technique for Fraud Detection. Lecture Notes in Computer Science, 341-348. <doi:10.1007/978-3-540-77046-6_43>

Pasha, M., Fatima, M., Dogar, A. M., & Shahzad, F. (2017, March). Performance Comparison of Data Mining Algorithms for the Predictive Accuracy of Credit Card Defaulters. Retrieved from <http://paper.ijcsns.org>

Sahin, Yusuf &Duman, Ekrem. (2011). Detecting credit card fraud by ANN and logistic regression. INISTA 2011 - 2011 International Symposium on INnovations in Intelligent SysTems and Applications. 10.1109/INISTA.2011.5946108.

Seeja, K. R., & Zareapoor, M. (2014). FraudMiner: A Novel Credit Card Fraud Detection Model Based on Frequent Itemset Mining. The Scientific World Journal, 2014, 252797. <http://doi.org/10.1155/2014/252797>

Zareapoor, M., & Shamsolmoali, P. (2015). Application of Credit Card Fraud Detection: Based on Bagging Ensemble Classifier. Procedia Computer Science, 48, 679-685. <doi:10.1016/j.procs.2015.04.201>

# Appendix:

### Detailed Exploratory Data Analysis

**Missing and Zero Values**

The data does not contain missing values and as such no imputation will be necessary.

**Descriptive Statistics**

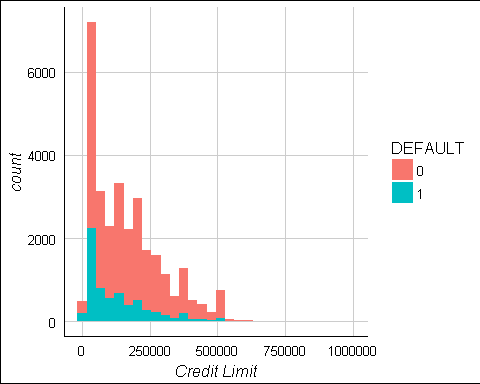
Descriptive statistics was performed for all predictor and response variables to explore the data.

**Analysis of Variables**

##   
## -2 -1 0 1 2 3 4 5 6 7 8   
## 2759 5686 14737 3688 2667 322 76 26 11 9 19

**LIMIT\_BAL**

The majority of customers have lower credit limits, as such the distribution is right skewed



### SEX

The majority of customers are female. This variable can be made into a dummy variable.

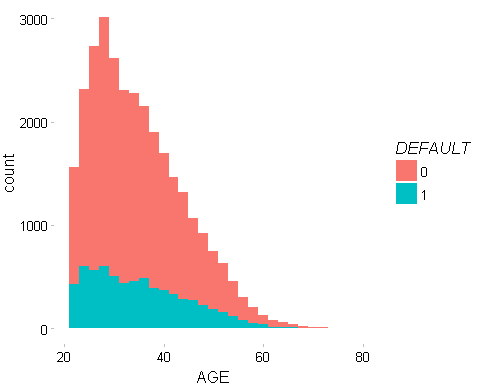
**EDUCATION**

The majority of customers went to university, there are very few in the other/unknown categories as well as an unknown 0 value and we will consider combining these values into dummy variables of College and Advanced Degree, with a 0 value in the College variable representing High School and all other possibilities.

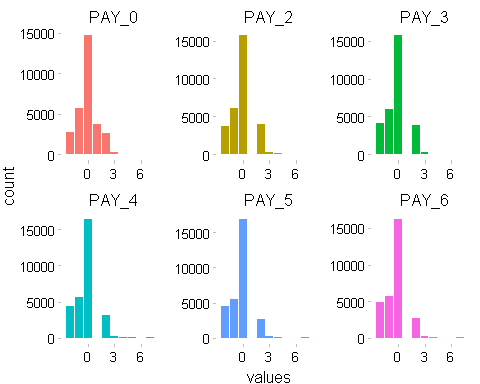
**MARRIAGE**

The majority of customers are single and the proportion of default payments appears to be higher for married individual. It appears that there are 0 values here which were not planned. It may be prudent to instead code this as a binary married variable.

**AGE**

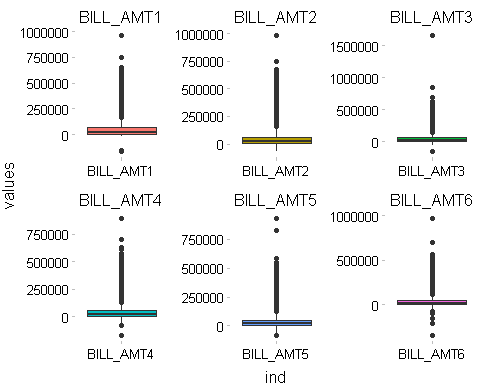
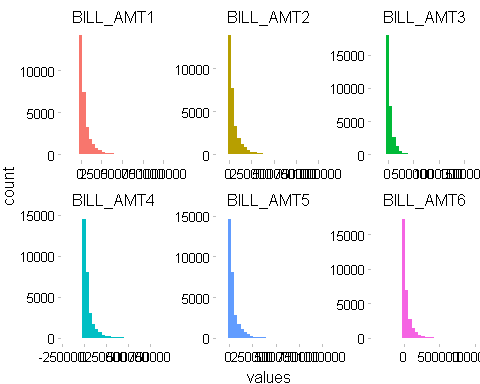
The distribution is right skewed. We can see that extremely young customers seem to have a higher proportion of defaults. 

**Repayment Status**

The vast majority of clients are on time or ahead of payments. The number of extremely late payments in the latter months are more infrequent. We can presume that this list only contains customers whose accounts have not yet been charged off, for which we would expect no future payments. Customers with extremely late repayment statuses 6 months ago have likely already been charged off. It is somewhat surprising that PAY\_0, PAY\_2 and PAY\_3 contain lower frequencies of extremely late payment statuses.

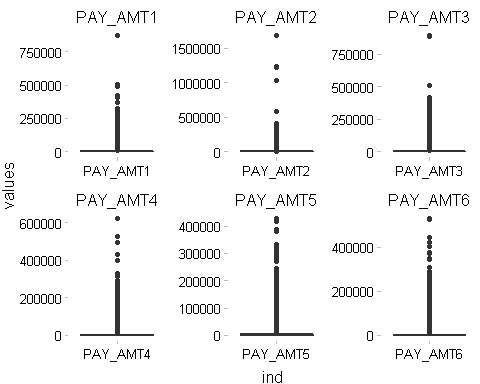
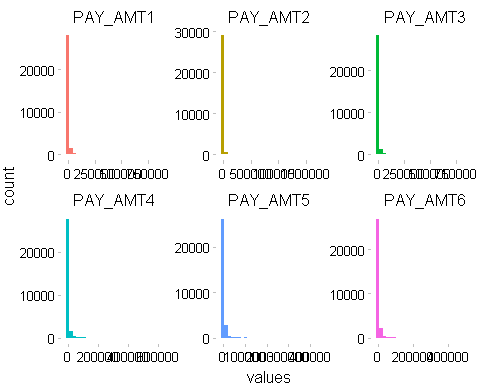
**Bill Amount**

We appear to have some negative values for bill amount. This likely represents overpayment by the customer and is not problematic. The distributions are similar.

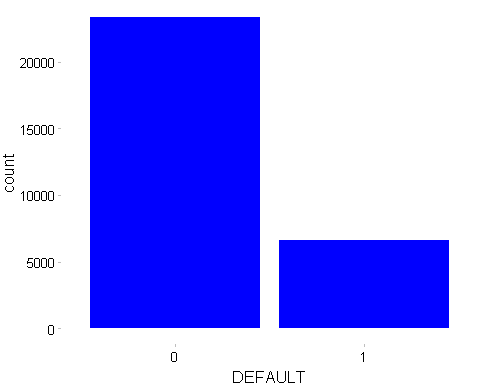


**Pay Amount**

The majority of payments are small with some rather large outliers.



**DEFAULT**

As we would expect the majority of customers are not in defualt of their next payment. 

**Data Manipulation**

As discussed previously, we will combine values 4, 5 and 6 in education to create an all encompassing other category. We will also change the unknown value of 0 in MARRIAGE to 3(other)

**Recode Predictors**

### Change values to factors

## LIMIT\_BAL AGE PAY\_0 PAY\_2 PAY\_3   
## "numeric" "integer" "integer" "integer" "integer"   
## PAY\_4 PAY\_5 PAY\_6 BILL\_AMT1 BILL\_AMT2   
## "integer" "integer" "integer" "numeric" "numeric"   
## BILL\_AMT3 BILL\_AMT4 BILL\_AMT5 BILL\_AMT6 PAY\_AMT1   
## "numeric" "numeric" "numeric" "numeric" "numeric"   
## PAY\_AMT2 PAY\_AMT3 PAY\_AMT4 PAY\_AMT5 PAY\_AMT6   
## "numeric" "numeric" "numeric" "numeric" "numeric"   
## DEFAULT MARRIED MALE EDU\_COLLEGE EDU\_ADV\_DEGREE   
## "integer" "numeric" "numeric" "numeric" "numeric"