Statistical Modeling to Predict Credit Approval

By

Aaron Beach, Bharadwaj Kopparthi, Sidra Husain, Trang Doan, Xiaojie Yu

Abstract - Credit decisions influence the daily life of individuals as well as the economic conditions of a nation. It may seem mysterious as to how the decisions are made to our credit card or loan applications. However, when we further look into the information that is required from us when applying for credit cards or loans, we understand that they are correlated with our incomes, debts, and so on. But this understanding is more like viewing a leopard through a tube - we can only see part of the whole picture at any one time. In this report, our group utilizes analytic tools to systematically evaluate what are the key factors in deciding credit approvals and how these factors influence credit decisions. We use multiple statistical models, including principal component analysis, linear regression, logistic regression, and neural networks. We identify several factors that are statistically significantly associated with credit decisions. These factors include PriorDefault, Employment status, Income, Credit Score, Years Employed, and Debt. We also demonstrate that our variable selection models coupled with principal component analysis narrow the variables down to the ones that are most likely to impact on credit decisions. At last, we illustrate that both logistic regression and Neural Network models provide high accuracy in predicting credit approval in the dataset we used.

I. INTRODUCTION

Credit decisions play a key role in the financial system, not only serving as the foundation for lending practices and capital allocation, but also as the engine that boosts economic growth. Whether made by financial institutions, credit rating agencies, or individual lenders, credit decisions involve evaluating the creditworthiness and risk profile associated with borrowers, and deciding whether or not to loan money to the applicant. Creditors often perform statistical analysis to investigate and predict the probability of a borrower to fulfill the agreed obligation and pay back the loan balance on time. An effective credit analysis method helps mitigate the risk of non-payment, reduces the prevalence of late payments or defaults, and contributes to a positive impact on a company's cash flow, profitability, and investment decisions.

During the last decade, the rapid advancements in technology have transformed various industries, and the field of credit analysis is no exception. Technological innovations have revolutionized the way credit analysis is conducted, improving efficiency, accuracy, and decision-making. In particular, numerous researchers have studied the application of data-mining tools to enhance the credit approval process. Some researchers focus on increasing prediction accuracy, while others aim to minimize computing times and other opportunity costs without compromising other financial metrics.

The primary objective of this analysis is to understand the application of data mining techniques in the credit approval process and to apply this knowledge in generating analytical models to predict credit approval decisions. The approval process is a matter of predicting probabilities and deciding which threshold to apply to optimally categorize a borrower as too risky or not to lend to. This lends itself naturally to predicting approval using a logistic regression model.

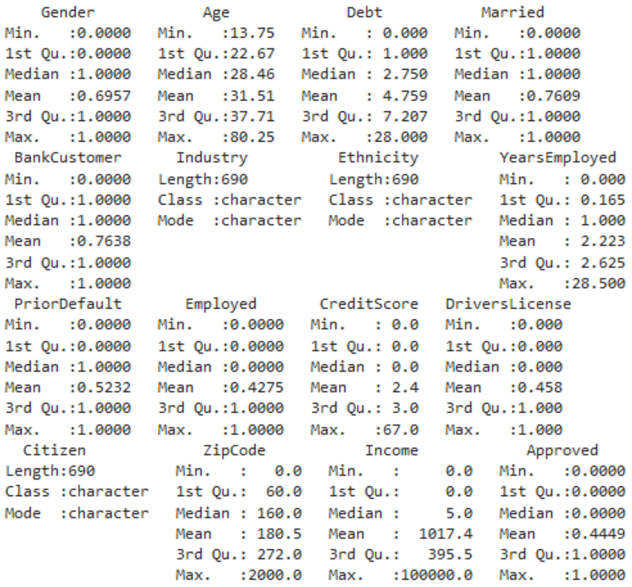
However, in recent years, financial institutions have been opting to use more advanced machine learning models due to their accuracy and ability to handle more complex data [1]. Neural networks are a type of machine learning model that have been shown to provide better accuracy in assessing risk, but have historically been extremely difficult to explain their inner workings. One key component in lending is how to identify in the borrowers what factored into an application denial. Researchers have been able to develop tools that help solve this problem of explaining the results of the model, which eliminates this barrier of communication that would otherwise keep a superior model from being utilized.

In the scope of this report, we not only analyze the credit decision process, but also compare two different models in predicting the outcomes. Specifically, we evaluated logistic regression with neural networks. We used variable selection methods to identify the important factors and further compared the two models using these variables as well as principal component analysis alone and in combination with these variables, to see which model and combination of inputs provides the best results.

II. EXPLORATORY ANALYSIS OF DATASET

1. *Variable Analysis*

Using the Credit Approval data set obtained from the UC Irvine Machine Learning Repository, we first conducted analysis on the variables to have a basic idea of the data. There are overall seventeen variables in the data set used for this analysis, with their summaries displayed below in Figure 0. In our initial analysis we have included ZipCode and Industry, though the finalized models do not include these variables. The "ZipCode" variable seems statistically significant in our initial analysis, but it is not real data so we decided to exclude it from our finalized models since we would not be able to draw meaningful conclusions from it when explaining our model. For the "Industry" variable, we have included it in the preliminary data analysis but we decided to exclude it from subsequent modeling for two reasons: first, there are many different components of this variable, meaning that we would have to create a large number of dummy variables that would make our model difficult to explain; second, further review of its values identifies that they are not representative or inclusive of the major industries in the US:



```
     Gender            Age             Debt            Married      
 Min.   :0.0000   Min.   :13.75   Min.   : 0.000   Min.   :0.0000  
 1st Qu.:0.0000   1st Qu.:22.67   1st Qu.: 1.000   1st Qu.:1.0000  
 Median :1.0000   Median :28.46   Median : 2.750   Median :1.0000  
 Mean   :0.6957   Mean   :31.51   Mean   : 4.759   Mean   :0.7609  
 3rd Qu.:1.0000   3rd Qu.:37.71   3rd Qu.: 7.207   3rd Qu.:1.0000  
 Max.   :1.0000   Max.   :80.25   Max.   :28.000   Max.   :1.0000  
  BankCustomer      Industry          Ethnicity         YearsEmployed   
 Min.   :0.0000   Length:690         Length:690         Min.   : 0.000  
 1st Qu.:1.0000   Class :character   Class :character   1st Qu.: 0.165  
 Median :1.0000   Mode  :character   Mode  :character   Median : 1.000  
 Mean   :0.7638                                         Mean   : 2.223  
 3rd Qu.:1.0000                                         3rd Qu.: 2.625  
 Max.   :1.0000                                         Max.   :28.500  
  PriorDefault       Employed       CreditScore    DriversLicense 
 Min.   :0.0000   Min.   :0.0000   Min.   : 0.0   Min.   :0.000  
 1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.: 0.0   1st Qu.:0.000  
 Median :1.0000   Median :0.0000   Median : 0.0   Median :0.000  
 Mean   :0.5232   Mean   :0.4275   Mean   : 2.4   Mean   :0.458  
 3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.: 3.0   3rd Qu.:1.000  
 Max.   :1.0000   Max.   :1.0000   Max.   :67.0   Max.   :1.000  
   Citizen             ZipCode           Income            Approved     
 Length:690         Min.   :   0.0   Min.   :     0.0   Min.   :0.0000  
 Class :character   1st Qu.:  60.0   1st Qu.:     0.0   1st Qu.:0.0000  
 Mode  :character   Median : 160.0   Median :     5.0   Median :0.0000  
                    Mean   : 180.5   Mean   :  1017.4   Mean   :0.4449  
                    3rd Qu.: 272.0   3rd Qu.:   395.5   3rd Qu.:1.0000  
                    Max.   :2000.0   Max.   :100000.0   Max.   :1.0000  
```

FIGURE 0

Variable Summary

Our next step in the exploratory data analysis was to check for missing or null values to determine whether any data imputation was required. For our data set, we did not identify any missing or null values for the variables. Hence, no data imputation was performed for further analysis. Additionally, we identified our dataset as a balanced dataset where the ratio of Approved to Decline is about equal to 1.

Plots of individual variables were done to inspect visually whether and assess which variable influences application approval (See Appendix A for full visualization). The most standout factor for us is how prior default impacted the approval decision. More than 90% of applications with default history get a denial (Figure 1). This is consistent with the findings of our later analysis that PriorDefault is one of the key factors in predicting credit decisions.

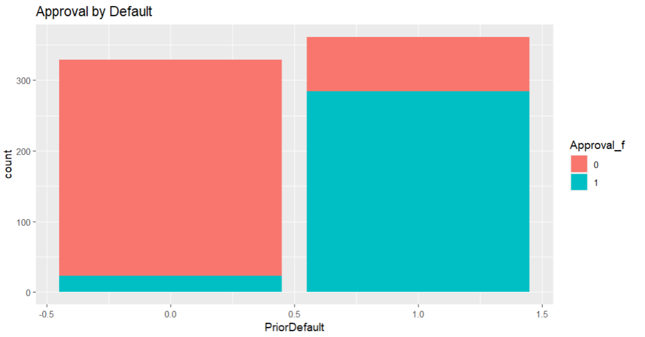


FIGURE 1

PriorDefault vs. Approval

Next, we investigated how credit score is related to income and the years of employment for those who were approved for their credit applications and for those who were not. The reason for doing this analysis is that Income is an important factor in determining a person's credit rating and that years of employment not only impact income but also establish an earning history, both of which are critical in determining credit approval. Figures 2 and 3 below indicate a strong correlation between Income/Age with CreditScore. Specifically, the green data points represent individuals that are approved for credit applications and red data points are for individuals who are not approved for credit applications. As we expected, individuals with higher income and better credit scores are more likely to get approved. Almost all points in Figure 2, quadrant I are green, while quadrant II, where both Income and CreditScore are in the lower ranges, contains more red data points.

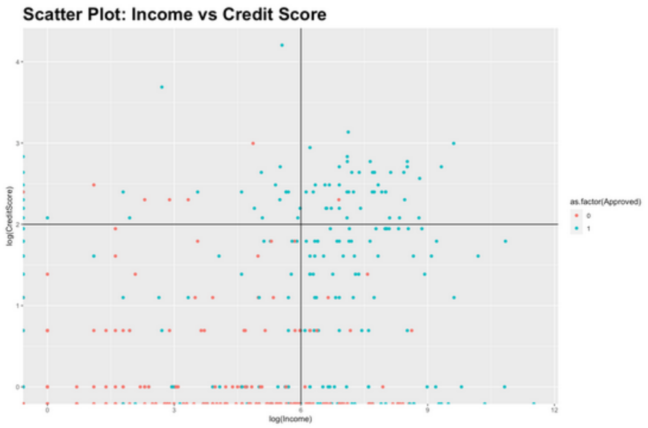


FIGURE 2
Scatter plot of Income vs Credit Score

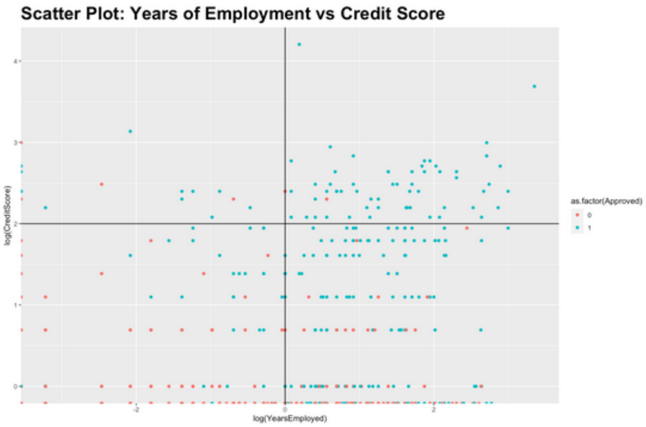


FIGURE 3
Scatter plot of Age vs Credit Score

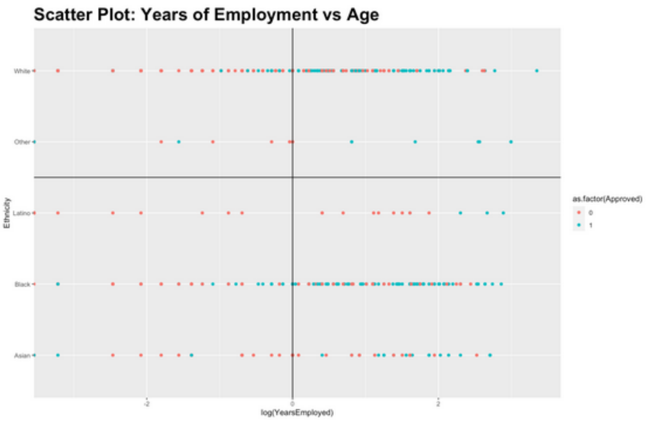


FIGURE 4
Scatter plot of Years of Employment vs Age

Similar conclusions can be drawn from Figure 3 & 4 as well. Applicants whose length of employment is longer usually are further advanced in their career path and likely will have higher incomes. If they also maintain their credit score, they highly likely belong to quadrant I in Figure 3, where all data points are green.

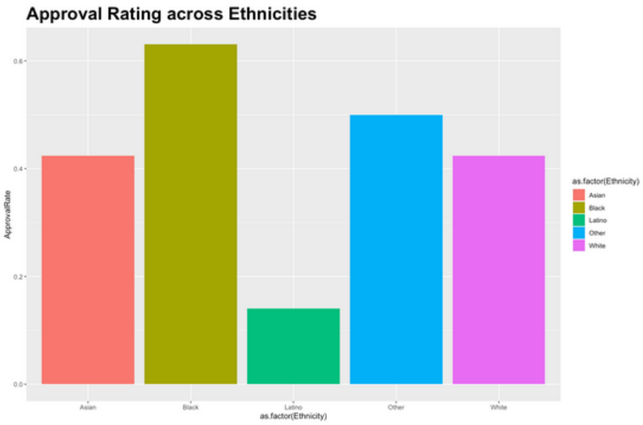
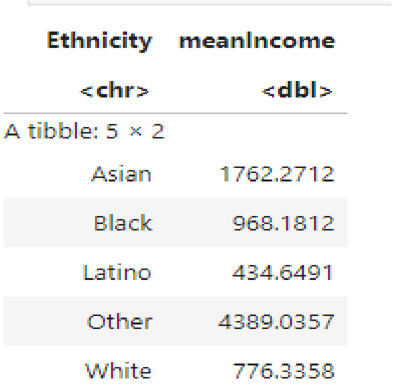


FIGURE 5
Histogram of Approval Rating based on Ethnicities

In this data set, we have four ethnic groups: White, Black, Asian, Latino, as well as a catchall group, Others. As shown, the Latino group has the lowest approval rate (< 15%). A famous study by UC Berkeley where researchers look into residential loans issued from 2008 to 2015 concluded that there is significant discrimination by both face-to-face and algorithmic lenders [2]. However, when we look into other factors among groups to compare, we see that Latino applicants also have the lowest average income. Since income will be an important factor in the credit decision process (explained later), we could not draw a conclusion as to whether or not there are inequality issues related to an applicant's ethnicity. We think the attributes in this data set are not presenting a true picture for issues related to ethnicity, or there might have been biases in the process of collecting income data.



Ethnicity	meanIncome
<chr>	<dbl>
A tibble: 5 × 2	
Asian	1762.2712
Black	968.1812
Latino	434.6491
Other	4389.0357
White	776.3358

FIGURE 6
Income vs. Ethnicity group

There are five continuous variables in our data set: Age, CreditScore, Income, and YearsEmployed. Plotting these variables in a histogram, we notice that they are all right skewed. Figure 7 is an example, showing the distribution of the Age variable. We will need to transform these variables using the logarithmic function (Figure 8). For each variable's distribution before and after log-transformation, see Appendix B.

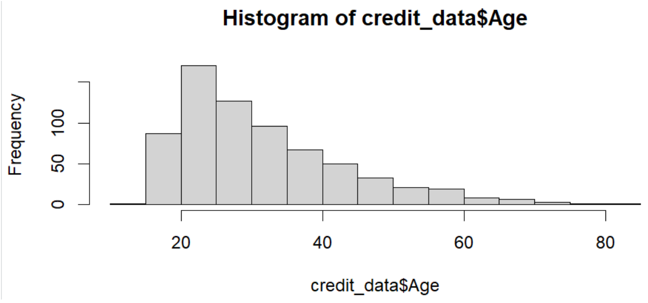


FIGURE 7
Age Variable Distribution

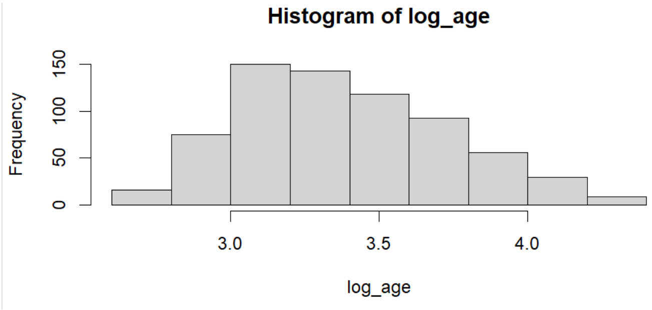


FIGURE 8
Log Transformed Age Variable Distribution

As part of the exploratory data analysis we also want to understand if any of the variables are correlated, aiming to eliminate multicollinearity. For this we created a subset by removing the categorical variables and we generated the following correlation matrix.

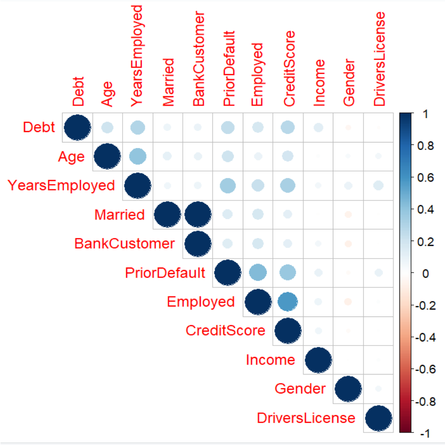


FIGURE 9
Correlation Matrix for the variables

From Figure 9 we can see that the Married and BankCustomer variables are highly correlated. We will address this issue in the subsequent variable selection step.

2. ***Variable Selection***

Our overall goal is to create the most promising model from our data set to predict whether or not a credit application should be approved. To prevent overfitting to random effects in our models we want to reduce the number of variables. We are also interested in reducing the number of variables so our model is easier to interpret and understand. For variable selection, we decided to use two approaches: a step-by-step analysis examining p-values through linear regression, and a classification and regression tree model. The idea behind using two different approaches is that both are simple and easy to interpret, and the tree based model mirrors human decision making and can accommodate qualitative variables. In the sections below we will first go through the statistical models for variable selection followed by the models for predictions, the neural network and the testing of the models.

Method 1: Variable Selection - Forward Stepwise Regression

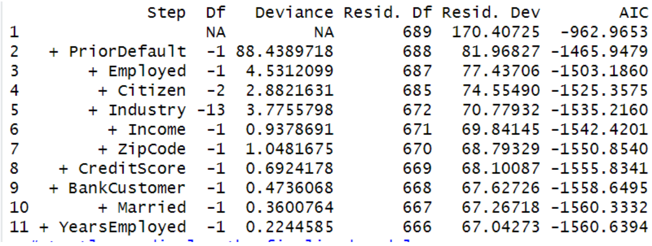
For determining which variables are statistically significant we will use stepwise regression. In stepwise regression, variables are added and removed to the regression model and the p-value is evaluated at each step [3]. For our analysis we have chosen forward stepwise regression. Using the clean dataset we first create a linear regression model for the intercept only with the regression function:

Approved = a_0

Next we create a linear regression model that includes all the variables:

Approved = a_0 + a_1*Prior Default + a_2 *Income +...

Next we run forward regression where we fit every variation of a one-predictor model followed by a two-predictor model and so on until no further reduction in AIC is observed.



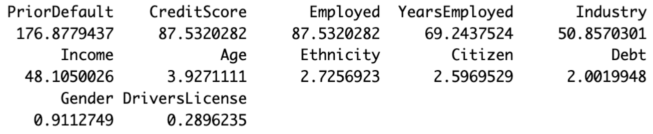
```
             Step  Df   Deviance Resid. Df Resid. Dev        AIC
1                  NA         NA       689  170.40725  -962.9653
2   + PriorDefault -1 88.4389718       688   81.96827 -1465.9479
3       + Employed -1  4.5312099       687   77.43706 -1503.1860
4        + Citizen -2  2.8821631       685   74.55490 -1525.3575
5       + Industry -13  3.7755798       672   70.77932 -1535.2160
6         + Income -1  0.9378691       671   69.84145 -1542.4201
7        + ZipCode -1  1.0481675       670   68.79329 -1550.8540
8    + CreditScore -1  0.6924178       669   68.10087 -1555.8341
9   + BankCustomer -1  0.4736068       668   67.62726 -1558.6495
10       + Married -1  0.3600764       667   67.26718 -1560.3332
11 + YearsEmployed -1  0.2244585       666   67.04273 -1560.6394
```

FIGURE 10
Forward Stepwise Regression Ranking of Variables

Based on the results above we have a list of variables that are statistically significant in predicting credit approval. These variables are ranked in their order of significance: PriorDefault, Employed, Citizen, Income, CreditScore, BankCustomer, Married and YearsEmployed.

Method 2: Variable selection - Classification and Regression Tree (CART)

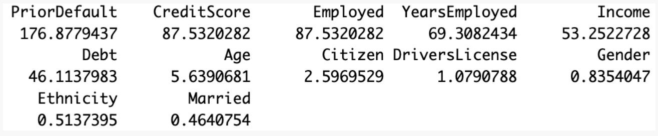
CART is another simple but powerful model when it comes to analyzing data. The benefit of CART is its ability to generate explainable, easy-to-understand models without the need for data preparation. The downside of CART is overfitting, thus pruning is usually recommended. However, for the purpose of analyzing important factors, we decide to utilize the simplistic result that the model generates. Below are the independent variables ranked based on their significance:



```
PriorDefault    CreditScore       Employed  YearsEmployed       Industry
 176.8779437     87.5320282     87.5320282     69.2437524     50.8570301
      Income            Age      Ethnicity        Citizen           Debt
  48.1050026      3.9271111      2.7256923      2.5969529      2.0019948
      Gender DriversLicense
   0.9112749      0.2896235
```

FIGURE 11
CART model showing importance variables

It was suspicious that Debt did not score highly in the CART analysis, and that Industry scored high. Rerunning the model, excluding Industry, we get the following results:



```
PriorDefault    CreditScore       Employed  YearsEmployed         Income
 176.8779437     87.5320282     87.5320282     69.3082434     53.2522728
        Debt            Age        Citizen DriversLicense         Gender
  46.1137983      5.6390681      2.5969529      1.0790788      0.8354047
   Ethnicity        Married
   0.5137395      0.4640754
```

FIGURE 12
CART model showing importance variables, excluding Industry

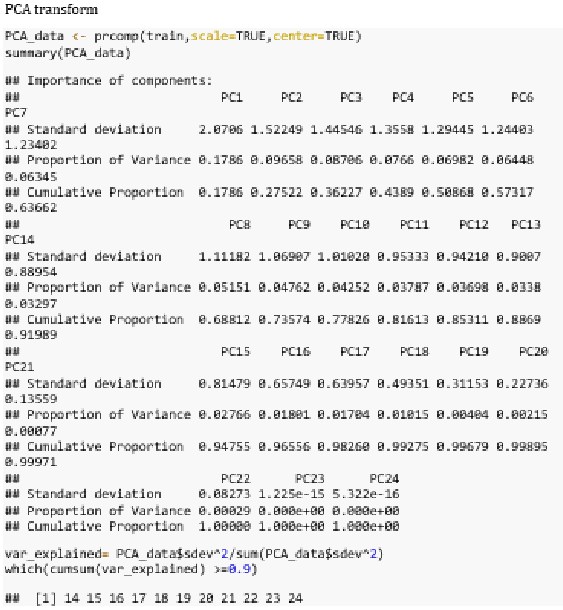
You will notice that Debt now has the 6th highest score and everything else is relatively the same. It seems that Industry, due to its large number of individual components, overweighed itself to earn a higher score, and pushed down the value of Debt. The variables PriorDefault, CreditScore, and Employed are still top three, which is consistent with the forward stepwise regression result. We moved forward using PriorDefault, CreditScore, Employed, YearsEmployed, Income, and Debt in our analysis.

3. ***Dimension Reduction using Principal Component Analysis***

Real life datasets often come in large sizes and contain noise and this dataset is no exception. During our examination of the independent variables, we ran a correlation test between variables and found out that two variables are significantly correlated: BankCustomer and Married (shown in Figure 9).

One of the main goals of regression analysis is to isolate the relationship between each independent variable and the dependent variable. When independent variables are not "independent", or multicollinearity exists, it makes the interpretation of a regression coefficient inaccurate. We can not confidently say that the regression coefficient represents the mean change in the dependent variable for each 1 unit change in an independent variable when holding all other independent variables constant [4].

Thus, we applied the PCA on the dataset and found that the first 15 components cover over 90% of the variance. Later on, we also see how transforming variables using PCA helps in reducing latency, which in real business applications is a significant factor to evaluate model performances. As business builds more sophisticated products, the requirement for faster data delivery becomes crucial and we definitely observe the differences when running the neural network model.



PCA transform

```
PCA_data <- prcomp(train,scale=TRUE,center=TRUE)
summary(PCA_data)

## Importance of components:
##                           PC1     PC2     PC3    PC4     PC5     PC6
PC7
## Standard deviation     2.0706 1.52249 1.44546 1.3558 1.29445 1.24403
1.23402
## Proportion of Variance 0.1786 0.09658 0.08706 0.0766 0.06982 0.06448
0.06345
## Cumulative Proportion  0.1786 0.27522 0.36227 0.4389 0.50868 0.57317
0.63662
##                            PC8     PC9    PC10    PC11    PC12   PC13
PC14
## Standard deviation     1.11182 1.06907 1.01020 0.95333 0.94210 0.9007
0.88954
## Proportion of Variance 0.05151 0.04762 0.04252 0.03787 0.03698 0.0338
0.03297
## Cumulative Proportion  0.68812 0.73574 0.77826 0.81613 0.85311 0.8869
0.91989
##                           PC15    PC16    PC17    PC18    PC19    PC20
PC21
## Standard deviation     0.81479 0.65749 0.63957 0.49351 0.31153 0.22736
0.13559
## Proportion of Variance 0.02766 0.01801 0.01704 0.01015 0.00404 0.00215
0.00077
## Cumulative Proportion  0.94755 0.96556 0.98260 0.99275 0.99679 0.99895
0.99971
##                           PC22      PC23      PC24
## Standard deviation     0.08273 1.225e-15 5.322e-16
## Proportion of Variance 0.00029 0.000e+00 0.000e+00
## Cumulative Proportion  1.00000 1.000e+00 1.000e+00

var_explained= PCA_data$sdev^2/sum(PCA_data$sdev^2)
which(cumsum(var_explained) >=0.9)

##  [1] 14 15 16 17 18 19 20 21 22 23 24
```

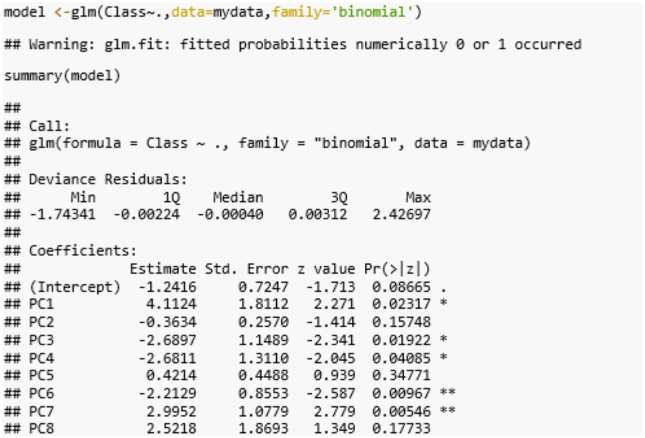
FIGURE 13
Principal Component Analysis Output

III. GENERATING ANALYTICS MODELS

1. ***Logistic Regression Model***

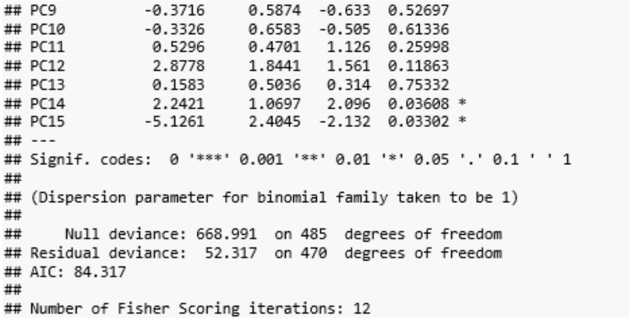
Given the nature of this dataset with many categorical variables and the binary outcomes (Approve/Deny), we decided to build a logistic regression model. The goal is to predict the probability of an application being approved using the logit function.

The model is created using the glm() function in R with binomial family. Using the predetermined variables, we come up with the model below. The summary of the model is presented below.



```
model <-glm(Class~.,data=mydata,family='binomial')

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

summary(model)

##
## Call:
## glm(formula = Class ~ ., family = "binomial", data = mydata)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.74341  -0.00224  -0.00040   0.00312   2.42697
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -1.2416     0.7247  -1.713  0.08665 .
## PC1           4.1124     1.8112   2.271  0.02317 *
## PC2          -0.3634     0.2570  -1.414  0.15748
## PC3          -2.6897     1.1489  -2.341  0.01922 *
## PC4          -2.6811     1.3110  -2.045  0.04085 *
## PC5           0.4214     0.4488   0.939  0.34771
## PC6          -2.2129     0.8553  -2.587  0.00967 **
## PC7           2.9952     1.0779   2.779  0.00546 **
## PC8           2.5218     1.8693   1.349  0.17733
```

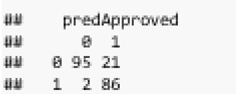
FIGURE 13a
Logistic Regression Model using PCA



```
## PC9          -0.3716     0.5874  -0.633  0.52697
## PC10         -0.3326     0.6583  -0.505  0.61336
## PC11          0.5296     0.4701   1.126  0.25998
## PC12          2.8778     1.8441   1.561  0.11863
## PC13          0.1583     0.5036   0.314  0.75332
## PC14          2.2421     1.0697   2.096  0.03608 *
## PC15         -5.1261     2.4045  -2.132  0.03302 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 668.991  on 485  degrees of freedom
## Residual deviance:  52.317  on 470  degrees of freedom
## AIC: 84.317
##
## Number of Fisher Scoring iterations: 12
```

FIGURE 13b
(contd) Logistic Regression Model using PCA

We set the cut-off at 0.5 for this model when using predict () for the testing dataset. The confusion matrix for this model is below. The accurate rate is 88.7%.



```
##     predApproved
##       0  1
##    0 95 21
##    1  2 86
```

FIGURE 14
Confusion matrix for the Logistic Regression Model using PCA

As mentioned before, the predicted value using logistic regression is based on the likelihood of an event and the binary outcome (Yes/No, Approved/Denied) is determined based on the cut-off. We have tried several different cut-off rates for this model and don't find significant increases in accuracy when changing the cut-off rate. However, the industry practice in determining cut-off rate is slightly different from academia. During different phases of the economy, the lenders might adjust the cut-off rate to balance between money supply and demand. The types of the loan and the borrowing amount might also impact the decision.

2. ***Neural Network Model***

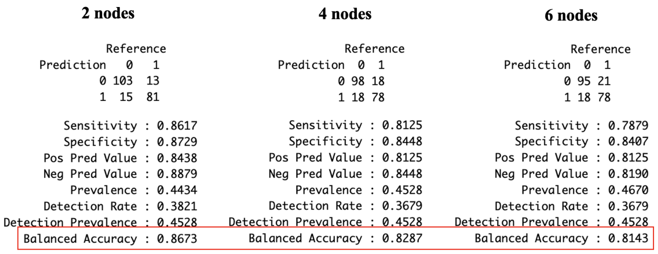
Neural networks are being employed by financial institutions more and more for assessing credit risk of individuals and thus aiding in decision-making for loan application approval. Neural networks have been shown to be more accurate than linear models in assessing credit risk, but they are also much more complex and difficult to interpret. One major component important to consumers in applying for loans is why they are denied, and until recently, neural networks have been considered "impenetrable 'black boxes'" that were impossible to explain [5]. Data scientists have been able to constrain these models and have them provide codes that are used to explain the credit decisions to customers.

We input the three sets of modified data into a neural net model. One note is that besides the variable selection and principal component analysis, a neural net is not required to have skewed data to be normalized [1]. Therefore, different from the logistic regression models, the Age, Debt, YearsEmployed, and Income variables are not log-transformed for the neural net models.

To run the neural net models, we used the neuralnet() function from the neuralnet library. Six total models, varying the number of layers and number of nodes, were tested to see which provided the best results for each of the three sets of inputs. The six variations are:

- 1 layer, 2 nodes
- 1 layer, 4 nodes
- 1 layer, 6 nodes
- 2 layers, 4 then 2 nodes
- 2 layers, 6 then 4 nodes
- 2 layers, 6 then 2 nodes

We tested these first on the six variables selected through the classification tree method, but the model had trouble converging on a solution if there was more than 1 layer with these inputs. Choosing between 1 layer with 2 nodes, 4 nodes, and 6 nodes, the accuracy was highest with 2 nodes:



2 nodes

```
             Reference
Prediction   0   1
         0 103  13
         1  15  81

        Sensitivity : 0.8617
        Specificity : 0.8729
     Pos Pred Value : 0.8438
     Neg Pred Value : 0.8879
         Prevalence : 0.4434
     Detection Rate : 0.3821
Detection Prevalence : 0.4528
  Balanced Accuracy : 0.8673
```

4 nodes

```
          Reference
Prediction  0  1
         0 98 18
         1 18 78

        Sensitivity : 0.8125
        Specificity : 0.8448
     Pos Pred Value : 0.8125
     Neg Pred Value : 0.8448
         Prevalence : 0.4528
     Detection Rate : 0.3679
Detection Prevalence : 0.4528
  Balanced Accuracy : 0.8287
```

6 nodes

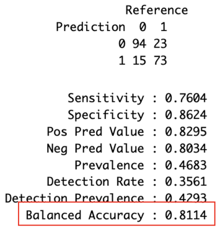
```
          Reference
Prediction  0  1
         0 95 21
         1 18 78

        Sensitivity : 0.7879
        Specificity : 0.8407
     Pos Pred Value : 0.8125
     Neg Pred Value : 0.8190
         Prevalence : 0.4670
     Detection Rate : 0.3679
Detection Prevalence : 0.4528
  Balanced Accuracy : 0.8143
```

FIGURE 15
Neural Network Model - Comparison

This was consistent through multiple iterations of splitting the data into training and testing subsets. The model using 1 layer and 2 nodes was also noticeably more efficient, so we analyzed the three sets of inputs using a model with 1 layer and 2 nodes.

*Neural Network using Six Significant Variable***s**

The model using the six determined significant variables provided good accuracy. Through ten iterations of splitting the data into training and testing subsets, the range of accuracies ranged from 0.8114 to 0.8881, with an average accuracy of 0.8507 and a standard deviation of 0.026. To compare models, we will report the lowest observed accuracy:

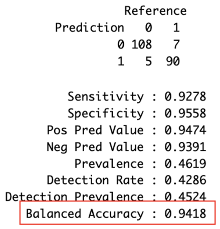


```
          Reference
Prediction  0  1
         0 94 23
         1 15 73

        Sensitivity : 0.7604
        Specificity : 0.8624
     Pos Pred Value : 0.8295
     Neg Pred Value : 0.8034
         Prevalence : 0.4683
     Detection Rate : 0.3561
Detection Prevalence : 0.4293
  Balanced Accuracy : 0.8114
```

FIGURE 16
Neural Network Model - 6 Variables

Neural Network using PCA

The model using principal component analysis provided very high accuracy. We did not limit the variables for this model except those of Industry and ZipCode. Through ten iterations of splitting the data into training and testing subsets, the range of accuracies went from 0.9418 to 0.9875, with an average accuracy of 0.9642 and a standard deviation of 0.017. You will notice that the range and standard deviation of accuracies are smaller, indicating the model is more consistent. It was also noticeable that the time it took to train the model using PC's was more quick than the one using variables. To compare models, we will report the lowest observed accuracy:

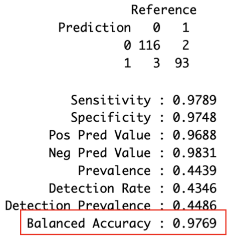


```
             Reference
Prediction   0   1
         0 108   7
         1   5  90

        Sensitivity : 0.9278
        Specificity : 0.9558
     Pos Pred Value : 0.9474
     Neg Pred Value : 0.9391
         Prevalence : 0.4619
     Detection Rate : 0.4286
Detection Prevalence : 0.4524
  Balanced Accuracy : 0.9418
```

FIGURE 17
Neural Network Model - PCA

Neural Network using Six Significant Variables & PCA

The model using both the six determined significant variables and then principal component analysis provided near perfect accuracy. Through ten iterations of splitting the data into training and testing subsets, the range of accuracies went from 0.9769 to 1.0000, with an average accuracy of 0.9903 and a standard deviation of 0.007. Along with the best average accuracy, this model has the best overall performance consistency. This model also trained very quickly. To compare models, we will report the lowest observed accuracy:



```
             Reference
Prediction   0   1
         0 116   2
         1   3  93

        Sensitivity : 0.9789
        Specificity : 0.9748
     Pos Pred Value : 0.9688
     Neg Pred Value : 0.9831
         Prevalence : 0.4439
     Detection Rate : 0.4346
Detection Prevalence : 0.4486
  Balanced Accuracy : 0.9769
```

FIGURE 18
Neural Network Model - 6 Variables and PCA

IV. CONCLUSION

In summary, we used multiple models to evaluate the decision-making process in credit approval analytically. We cleaned the dataset, transformed the variables that are not normally distributed, performed variable selection to reduce overfitting and the complexity of our models, and utilized logistic regression and neural network separately to predict credit approval. Interestingly, regardless of the models that we used for variable selection, the same factors are identified as being statistically significant in the credit decision process: PriorDefault, Employed, Income, CreditScore, YearsEmployed, and Debt. We proceeded our project with the assumption that these variables would be significantly associated with credit decisions, and were able to confirm this later using Stepwise Regressions and CART. Both logistic regression and neural networks generated promising results in predicting credit approval outcomes.

Logistic regression requires a clean set of data and adds a layer to the time needed to collect and analyze data, whereas a neural network is able to work with complex, unprocessed data. Intriguingly, it seems that both logistic regression and Neural Network produce outcomes with high accuracy. We believe that the results of these two models validate the results of each other. Since the Neural Network model requires less steps in data preparation, we prefer Neural Network over logistic regression. Actually, more and more institutions are now utilizing neural networks to improve their ability to handle complex projects with better accuracy. Our project confirmed that, at the least, a neural network's performance is better than a logistic regression, and at most, it can perform extremely well with some data cleaning and dimensional reduction. Ideally, we could apply our models to a larger dataset to evaluate their performance.see how it performs. And if with more resources, we could purchase an even larger set of data from a financial institution.

References

[1] Seetharaman, K. (2018, February 22). Financial Applications of Neural Networks. Blog. Retrieved from https://www.aspiresys.com/banking-financial-services.

[2] Bartlett,Robert, Morse,Adair, Stanton,Richard and Wallace,Nancy. (2019,November). Consumer-Lending Discrimination in the Fintech Era. Retrieved from http://faculty.haas.berkeley.edu/morse/research/papers/discrim.pdf?_gl=1*15rr2lu*_ga*MTMzMDc2OTExMi4xNjgwMDE3ODQ4*_ga_EW2RSBHHX6*MTY4OTg3ODc1NS4xLjAuMTY4OTg3ODc1NS4wLjAuMA..

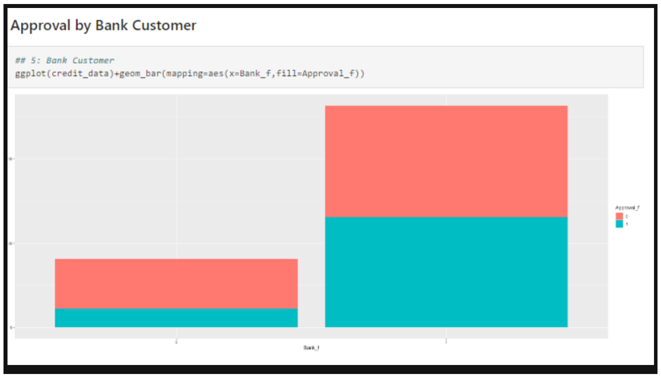
[3] Statology.org. (2019, April 27). A Complete Guide to Stepwise Regression in R. Retrieved from https://www.statology.org/residual-plot-r/

[4] Frost, J. (n.d.). Multicollinearity in Regression Analysis. Retrieved from https://statisticsbyjim.com/regression/multicollinearity-in-regression-analysis/

[5] McBurnett, M. (2020, December 16). How Neural Network Models Put Financial Services Within Reach. Retrieved from https://www.equifax.com/business/blog/-/insight/article/neural-network-models-put-financial-services-within-reach//

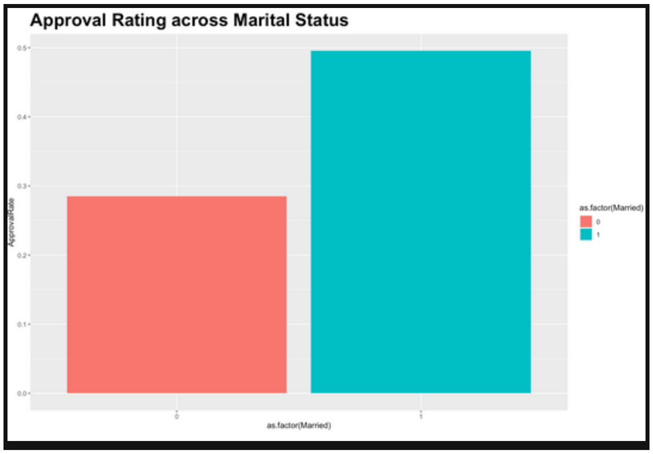
Appendix A

Visualization of individual variables's effect on approval .

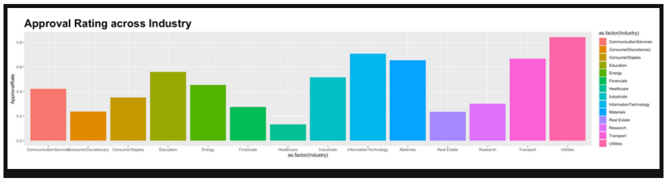


Approval by Bank Customer

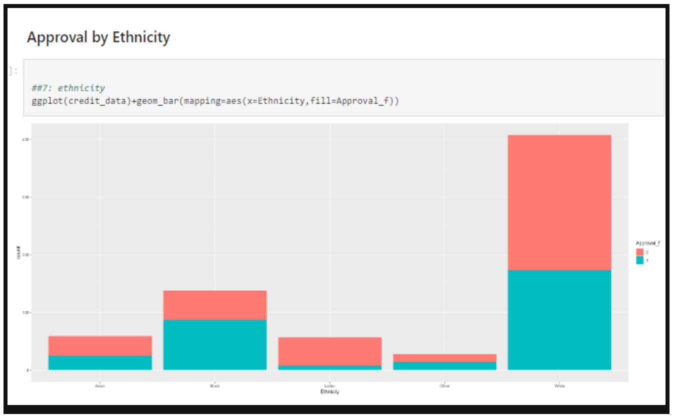
```
## 5: Bank Customer
ggplot(credit_data)+geom_bar(mapping=aes(x=Bank_f,fill=Approval_f))
```



Approval Rating across Marital Status

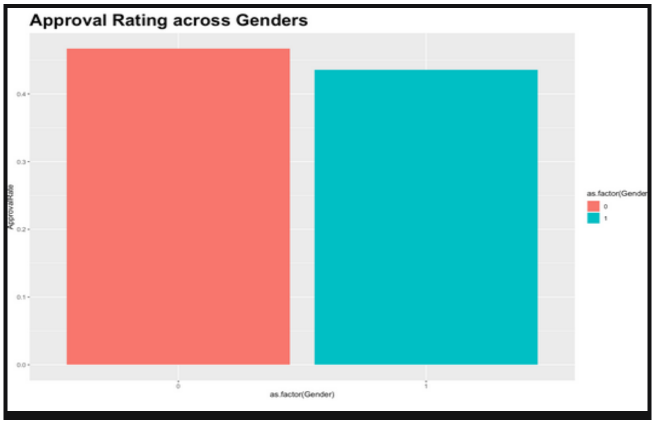


Approval Rating across Industry



Approval by Ethnicity

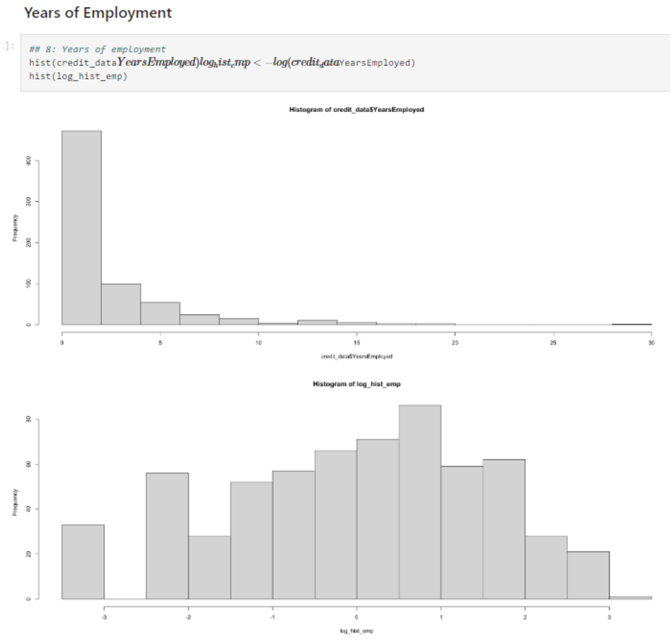
```
##7: ethnicity
ggplot(credit_data)+geom_bar(mapping=aes(x=Ethnicity,fill=Approval_f))
```



Approval Rating across Genders

Appendix B

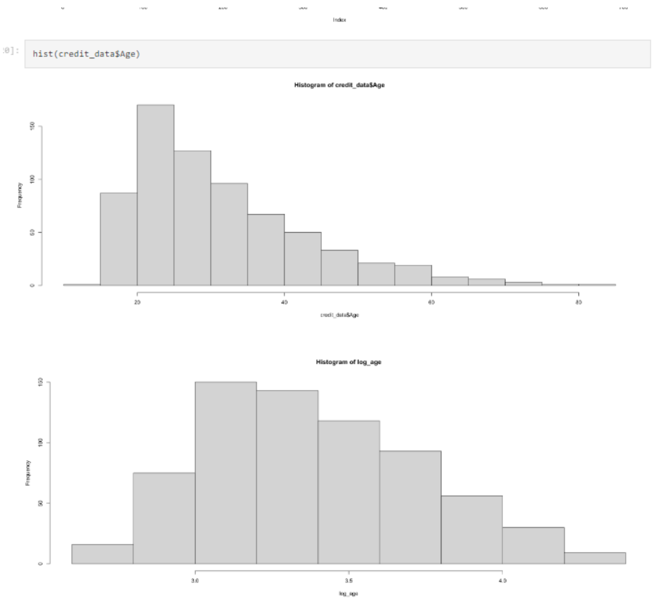
Continuous Variable Transform using Logarithm



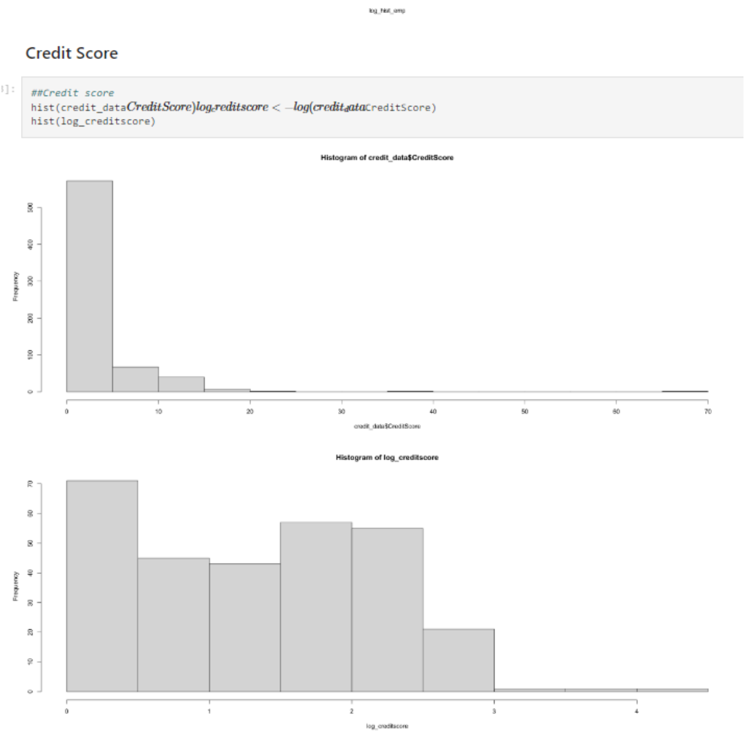
Years of Employment

```
## 8: Years of employment
hist(credit_dataYearsEmployed)log_hist_emp < −log(credit_dataYearsEmployed)
hist(log_hist_emp)
```

Histogram of credit_data$YearsEmployed
Frequency
0
100
200
300
400
0
5
10
15
20
25
30
credit_data$YearsEmployed
Histogram of log_hist_emp
Frequency
0
20
40
60
80
-3
-2
-1
0
1
2
3
log_hist_emp

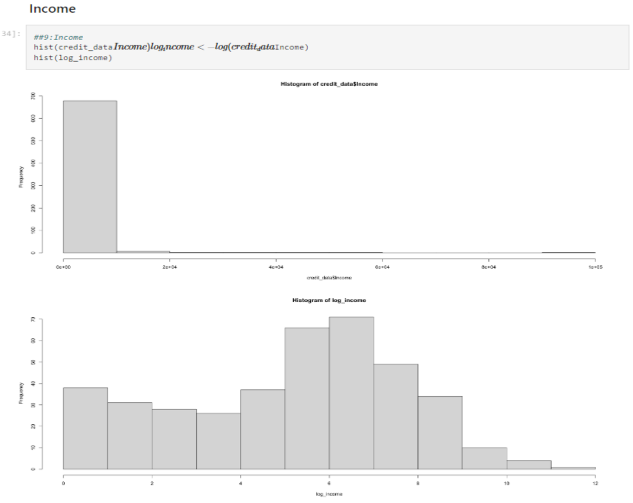


```
hist(credit_data$Age)
```



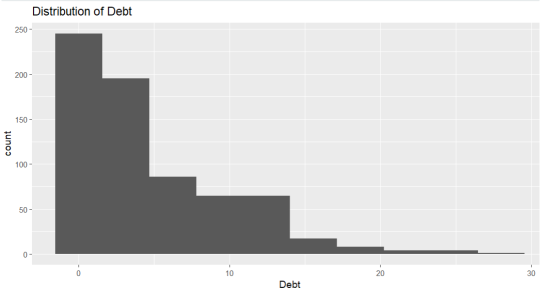
Credit Score

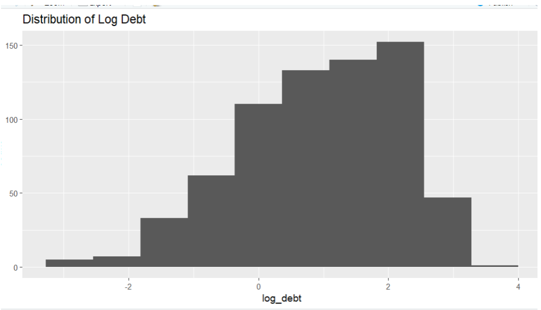
```
##Credit score
hist(credit_dataCreditScore)log_creditscore < −log(credit_dataCreditScore)
hist(log_creditscore)
```



Income

```
##9:Income
hist(credit_dataIncome)log_income < −log(credit_dataIncome)
hist(log_income)
```




Distribution of Log Debt
150
100
50
0
-2
0
2
4
log_debt