# Classification Model Comparison and Improvement Regarding Credit Risk

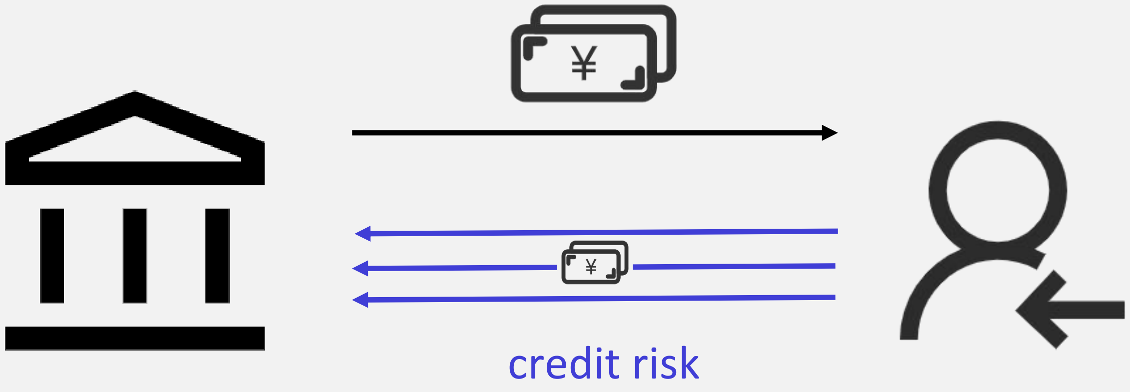
Simin Yu, Yuan Tan, Yuying Zhang, Kexin Sheng

ACFBP 2021

# contents

# PART 1
# Introduction

# 1.1 Background




Credit risk measurement is an important topic in the finance market since the precise prediction of borrowers' defaults can help banks or companies to maximize profits.

Mnay classification models have been developed. However, model performance depends on classifying accuracy, and there is not a single one that fits for all.

# 1.2 Literature Review

**Models applied to credit risk analysis**

Henley and Hand (1996): K-nearest-neighbourhood (KNN)

Farquad et al. (2011): PCA-SVM model, better performance compared to SVM  or PCA-Logistic Regression model alone

Lappas et al. (2021): Unsupervised ML combining with genetic algorithms

Ünvan (2019): Quantile Regression

Qasem et al. (2020): Extreme Learning Machine (ELM), better performance compared to naive Bayes, decision tree, and Multi-Layer Perceptron (MLP)

Due to space constraint, not all literature are presented.

# PART 2
# Data Description

# 2.1 Data Overview

Our research is based on simulation credit bureau data from *Kaggle*.

01

12 categories
32,573 data sets in total

02

Eliminated extreme data

03

# 2.2 Data Dimension

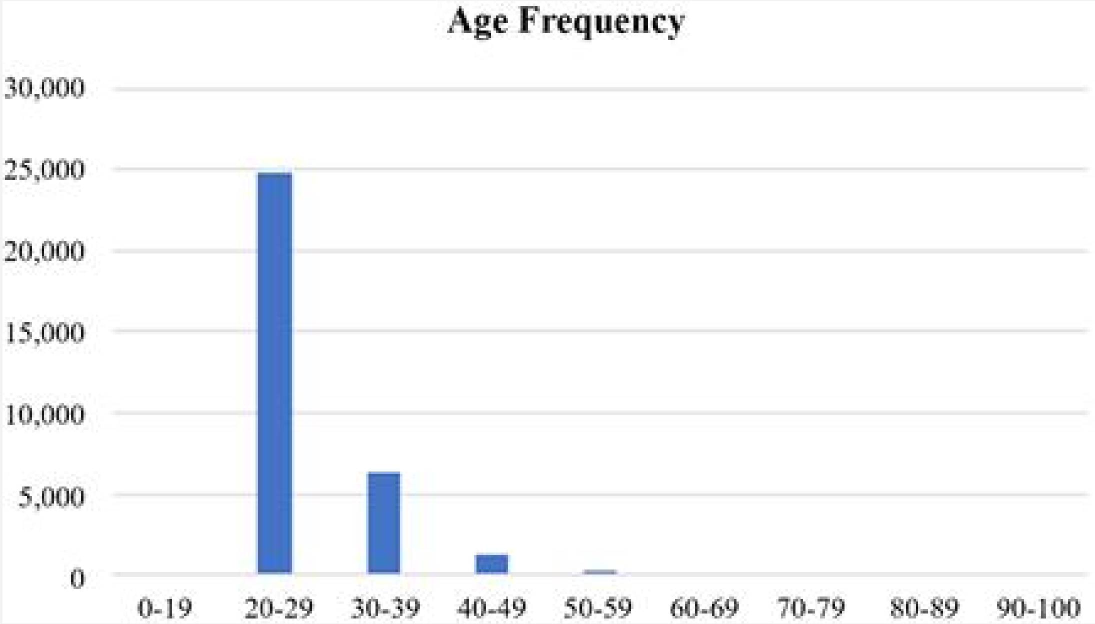
- Age
- Annual Income
- Loan amount
- Loan grade (A/B/C/D/E/F and A is the highest grade)
- Historical default (0-non default, 1-default
- Loan intent
- Interest rate
- Percent income
- Credit history length
- Employment length (in years)
- Home ownership
- Loan intent

**12**

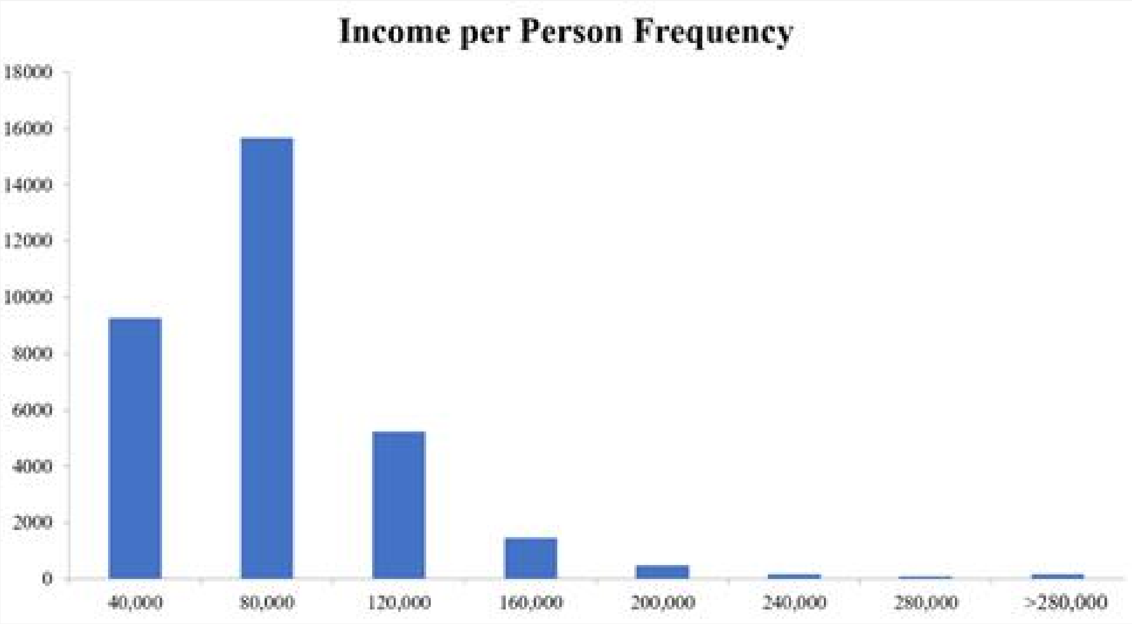
**dimensions**

# 2.3 Data Distribution
(Partly)

### Age

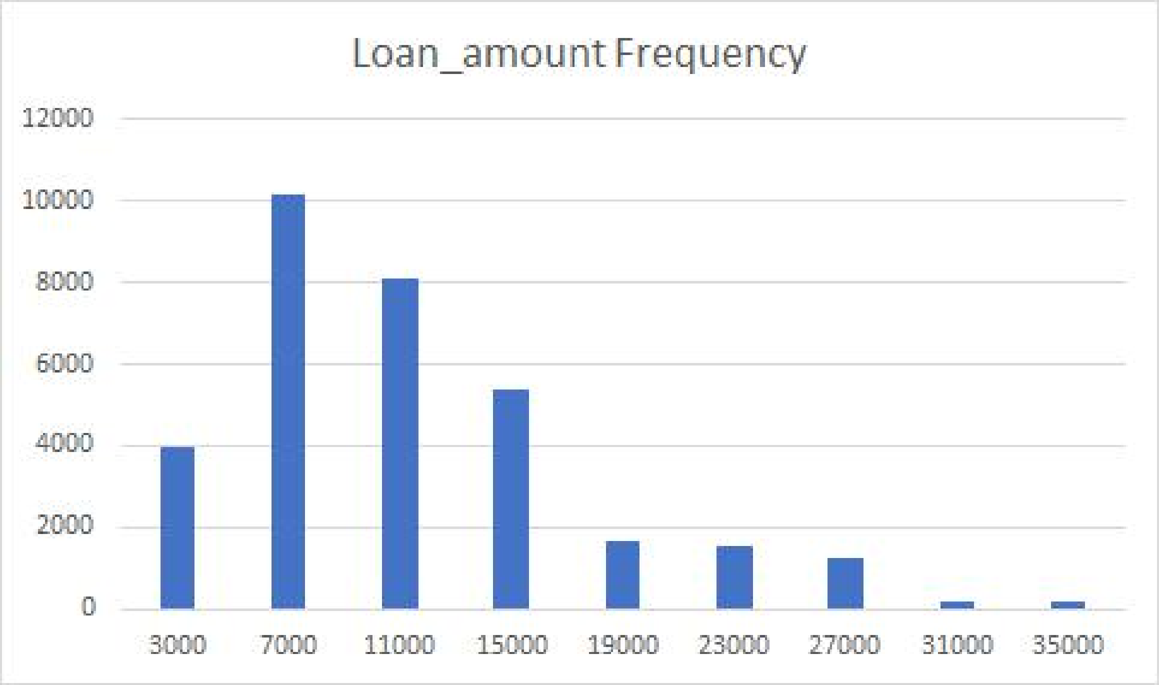



### Annual Income

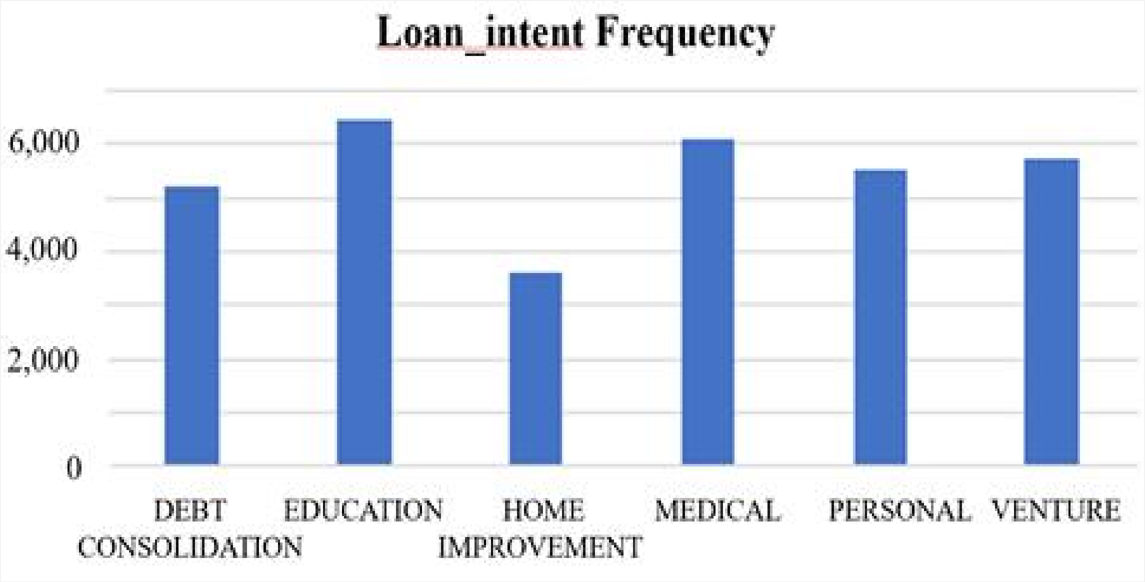


# 2.3 Data Distribution
(Partly)

## Loan Amount

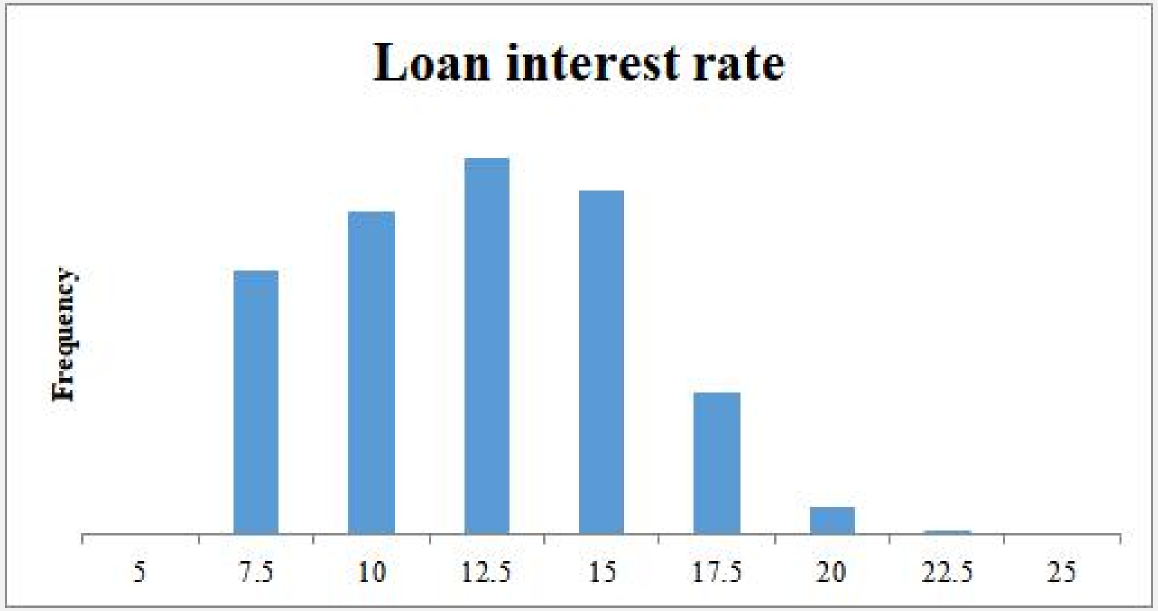



## Loan Intent

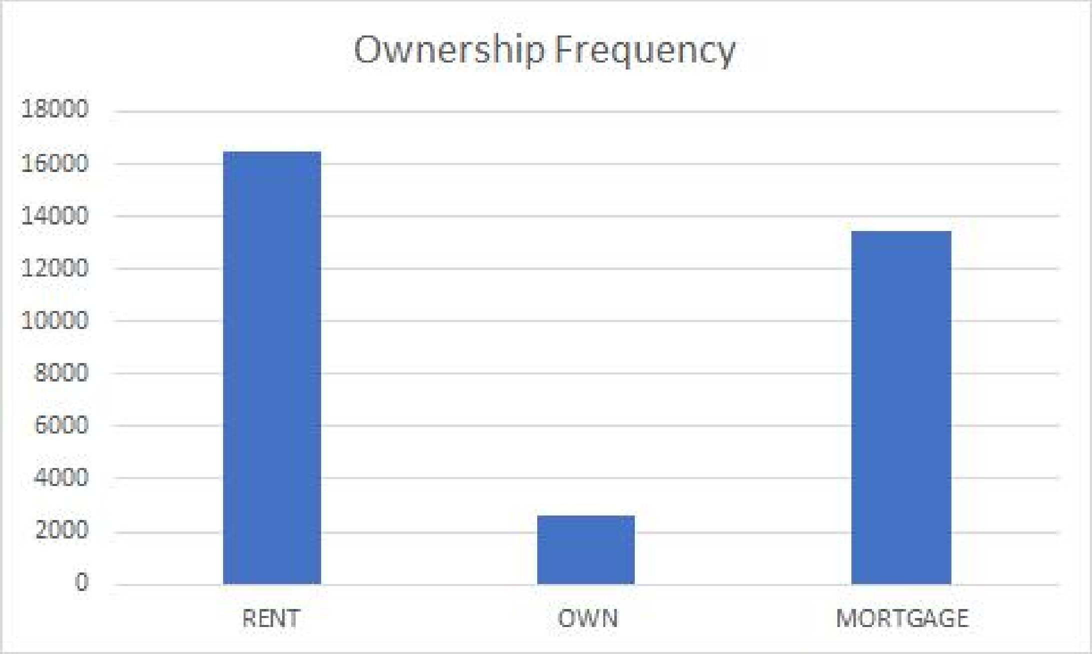


# 2.3 Data Distribution
(Partly)

### Loan Interest Rate

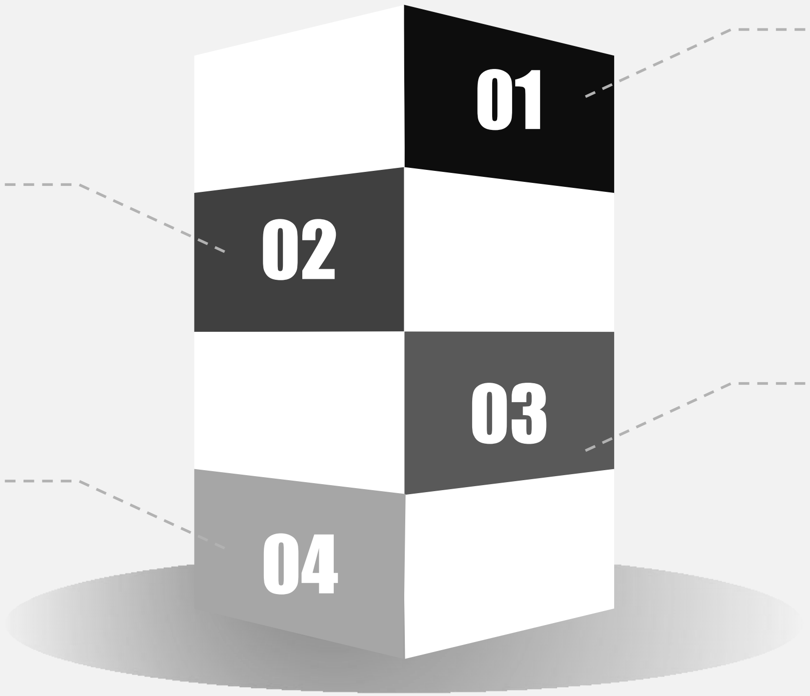



### Types of Ownership



# PART 3
# Model Analysis

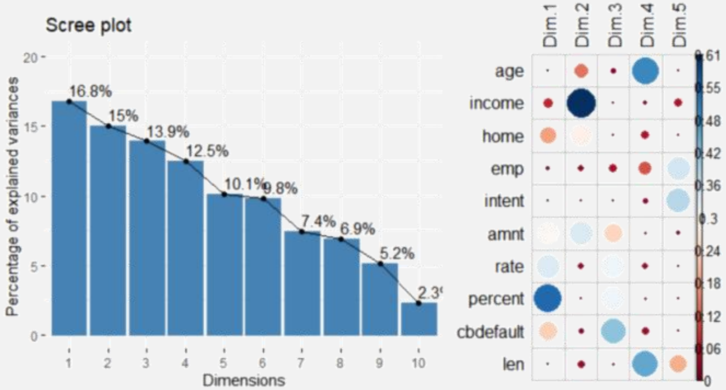
# 3.0 Data Preprocessing




Drop the rows with null values

Use Synthetic Data Generation to deal with the unbalanced data (22430 non-default and 6202 default)
→get 14411 non-default data and 14221 default data

Conduct Correlation Test and Principal Component Analysis, choose six principal components
→reduce dimension and perserve information




Divide the dataset into training and testing sets in a 4:1 scale
→avoid overfitting

# 3.1 Multiple Logisic Regression

## EXPLANATION

The probability of default is postlated to be a function of a set of regressors:

$$ln\left(\frac{p}{1-p}\right)=X\beta=\begin{pmatrix}1 & x_{11} & \cdots x_{k1}\\ \vdots & \vdots & \ddots\ \vdots\\ 1 & x_{in} & \cdots x_{kn}\end{pmatrix}\begin{pmatrix}\beta_0\\ \vdots\\ \beta_k\end{pmatrix}\Rightarrow p=\frac{1}{1+e^{-X\beta}}$$

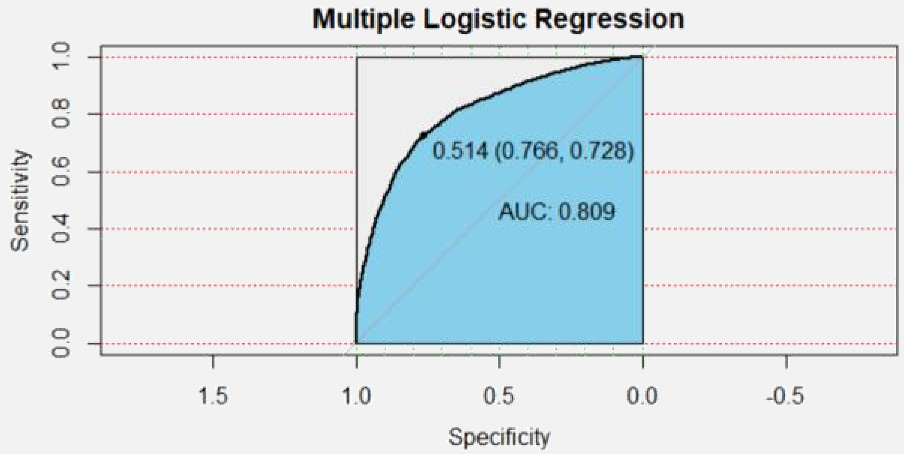
## RESULT

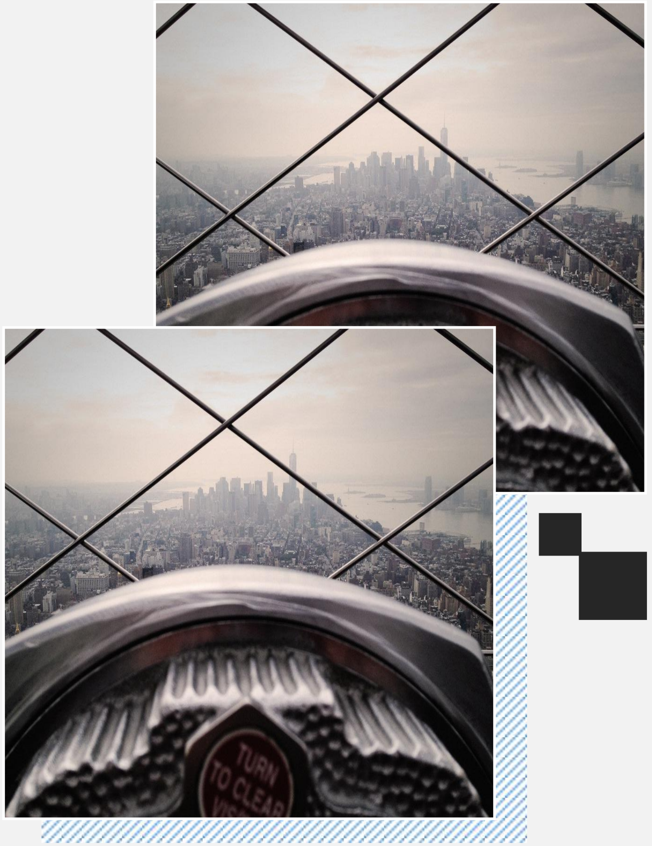
The result is:

$$ln\left(\frac{p}{1-p}\right)=-0.04+0.73R1-0.92R2+0.82R3+0.05R4-0.13R5-0.24R6$$

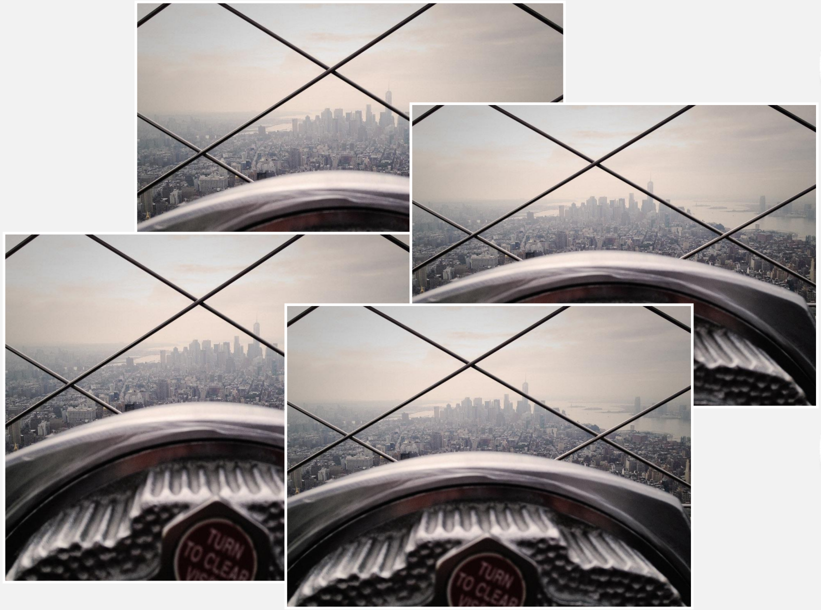
The mean square error on the training data is 0.176 and the mean square error on the testing data is 0.173.

The ROC curve on the testing data is shown below and AUC=0.809:



# 3.2 K-Nearest-Neighborhood (KNN)

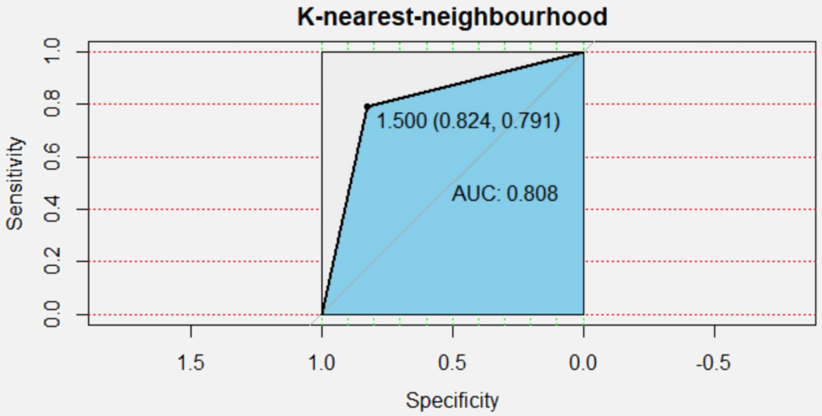


## EXPLANATION

The property of testing data is estimated according to the properties of the nearest k objects with weighted average method. The weight is inversely proportional to distance (here we use Euclidean Distance: $D(x,y) = \sqrt{\sum_{i=0}^{n}(x_i - y_i)^2}$).

## RESULT

To avoid underfitting and overfitting, parameter k is set as 15.
The accuracy on the training data is 0.8879 while the accuracy is 0.8074 on the testing data.
The ROC curve on testing data is shown on the left and AUC=0.808.




| train | 0 | 1 | test | 0 | 1 |
|---|---|---|---|---|---|
| 0 | 10383 | 1157 | 0 | 2329 | 497 |
| 1 | 1411 | 9955 | 1 | 606 | 2294 |

# 3.3 Support Vector Machine (SVM)

## EXPLANATION

SVM is a two-category model. For linear unseparated data, we use kernel function to map the points in low-dimensional space to higher-dimensional space where they are linear separated.

## RESULT

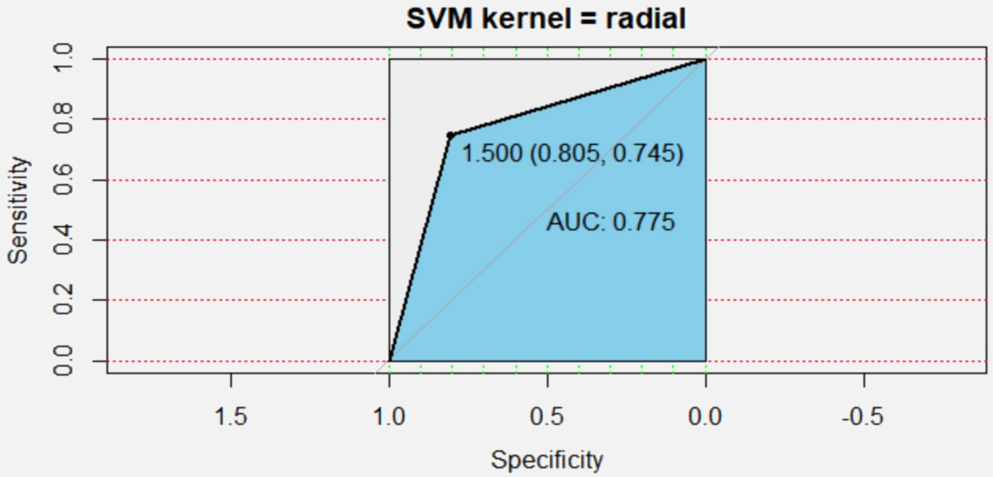
We use 4 different kernel functions including radial, polynomial, linear and sigmoid function to build the SVM model, to find that the radial function performs far better than others after an initial simple analysis.

Radial Basis Function: $K(v_1, v_2) = \exp(-\gamma||v_1 - v_2||^2)$

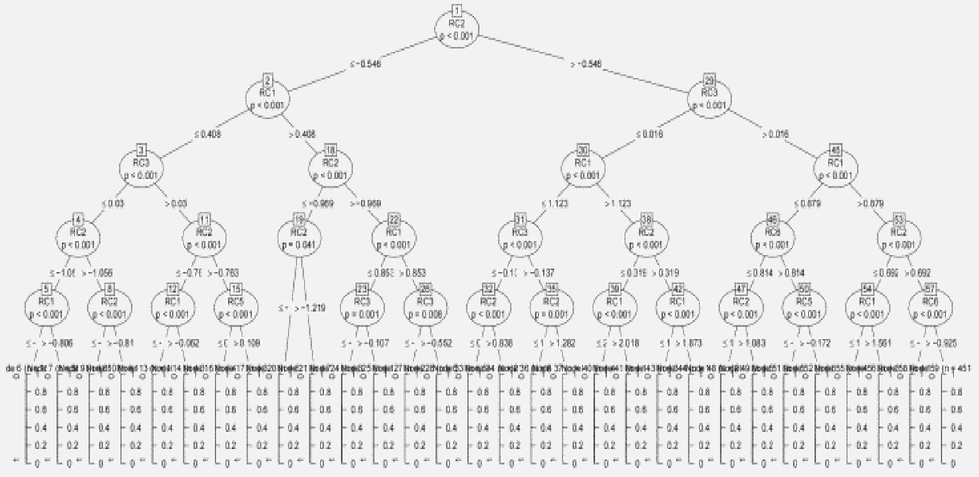
The accuracy on the training data is 0.7811 while the accuracy is 0.7745 on the testing data.

The ROC curve on testing data is shown on the right and AUC=0.775.

| train | 0 | 1 | test | 0 | 1 |
|---|---|---|---|---|---|
| 0 | 9275 | 2265 | 0 | 2274 | 739 |
| 1 | 2748 | 8618 | 1 | 552 | 2161 |



# 3.4 Decision Tree

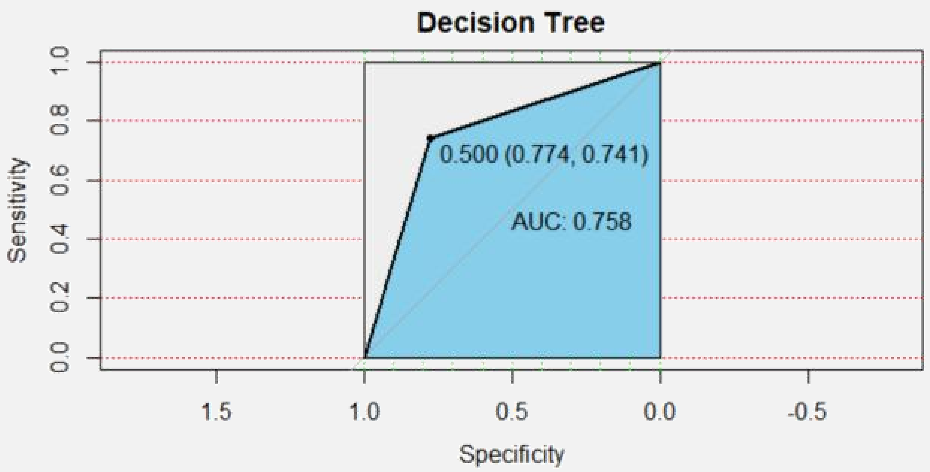


## EXPLANATION

Decision Tree is a model for classification that uses recursive segmentation of nodes using criteria like “gini impurity” or “information gain” to create tree models. It looks like a series of if-else statements arranged in tree form.

## RESULT

To avoid overfitting, we set max depth equals to 5.
The accuracy on training data is 0.7677, and the accuracy on the testing data is 0.7574.
The ROC curve on testing data is shown on the left and AUC=0.758.




| train | 0 | 1 | test | 0 | 1 |
|---|---|---|---|---|---|
| 0 | 9202 | 2949 | 0 | 2188 | 751 |
| 1 | 2372 | 8383 | 1 | 638 | 2149 |

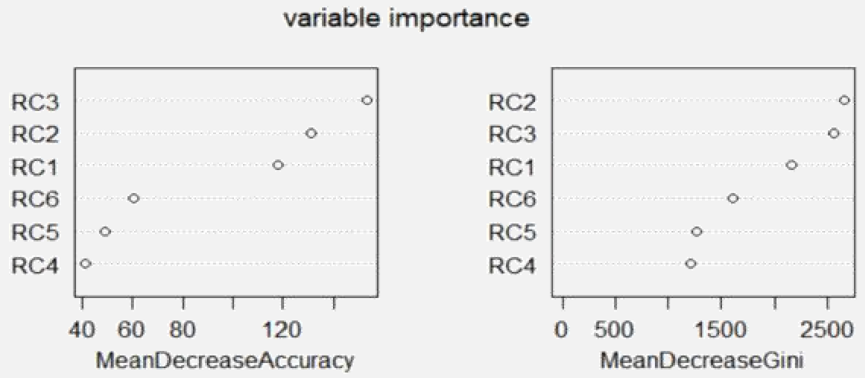
# 3.5 Random Forest

## EXPLANATION

Random Forest is composed of many decision trees.
The term "random" is embodied in two aspects:

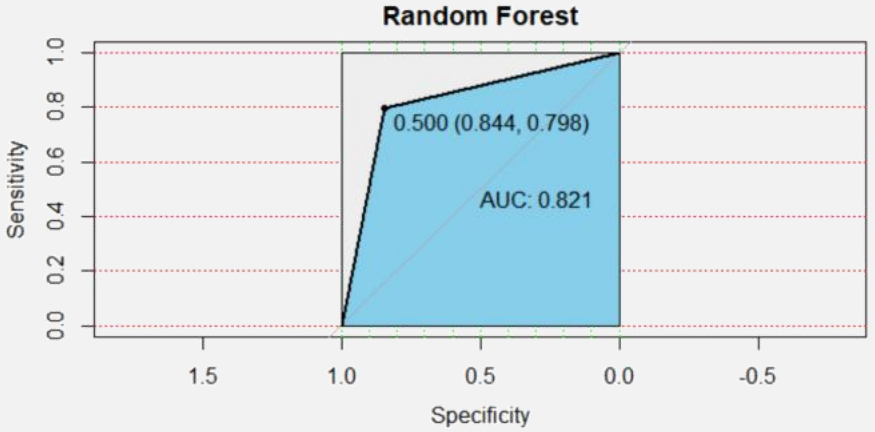
- If the taining data is of size N, for each tree, we randomly take N training samples from the data and put back. So each tree' s training set is different and contains duplicate sampls.
- We randomly select m subsets of features from M features (m<<M).

Each tree grows to its best without pruning. Finally the output category is the mode of all decision tree' s output.




## RESULT

We use a forest that consists of 100 trees. Among six principal components, RC3 is the most important variable in terms of the decrease of accuracy and RC2 is the most important with respect to the decrease of gini impurity.
The accuracy on training data is 1, and the accuracy on the testing data is 0.8206.
The ROC curve on testing data is shown on the right and AUC=0.821.




| train | 0 | 1 | test | 0 | 1 |
|---|---|---|---|---|---|
| 0 | 11574 | 0 | 0 | 2384 | 585 |
| 1 | 0 | 11332 | 1 | 442 | 2315 |

# 3.6 Model Comparison

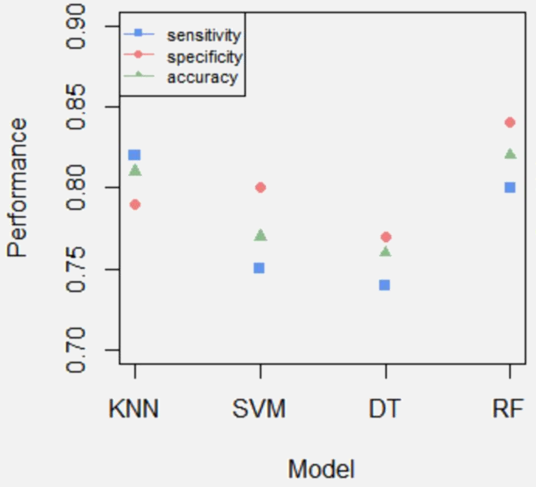
## Comparison Criteria

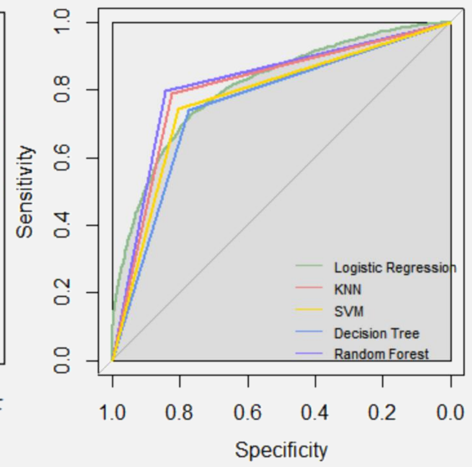
- Sensitivity represents the correctly predicted rate of observed positive results.
- Specificity indicates the ratio of observed negative results confused with the positive classification.
- Accuracy represents the correct proportion of the prediction.
- The receiver operating characteristic (ROC) curve and area under the curve (AUC) consider both sensitivity and specificity. The ROC with greater AUC indicates better performance.

## Conclusion

| | sensitivity | Specificity | accuracy |
|---|---|---|---|
| **KNN** | 0.82 | 0.79 | 0.81 |
| **SVM** | 0.75 | 0.80 | 0.77 |
| **Decision Tree** | 0.74 | 0.77 | 0.76 |
| **Random Forest** | 0.80 | 0.84 | 0.82 |

- KNN performs the best in terms of sensitivity, and in terms of specificity and accuracy, Random Forest performs the best. In general, random forest performance is quite superior.
- AUC of the five models are 0.809 (Multiple Logistic Regression), 0.808 (KNN), 0.775 (SVM), 0.758 (Decision Tree), and 0.821 (Random Forest), respectively. So Random Forest performs best,



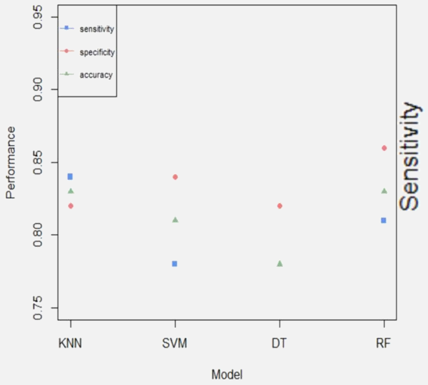
# 3.7 Model Analysis without PCA

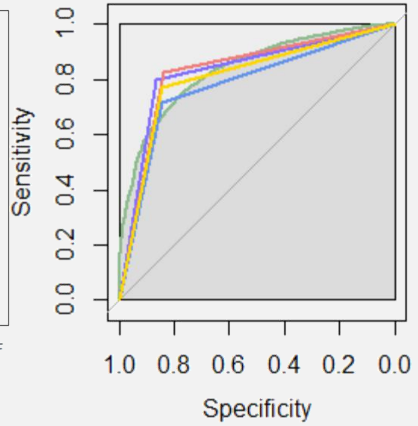
## Model Performance

In order to compare which variables are most important and predictive, we decide to do the model analysis without PCA.

| | sensitivity | specificity | accuracy |
|---|---|---|---|
| KNN | 0.84 | 0.82 | 0.83 |
| SVM | 0.78 | 0.84 | 0.81 |
| Decision Tree | 0.74 | 0.82 | 0.78 |
| Random Forest | 0.81 | 0.86 | 0.83 |

KNN performs the best in terms of sensitivity and accuracy, and in terms of specificity and accuracy, Random Forest performs the best.
AUC of the five models are 0.833 (Multiple Logistic Regression), 0.832 (KNN), 0.809 (SVM), 0.779 (Decision Tree), and 0.831 (Random Forest), respectively.
So Multiple Logistic Regression performs best, KNN and Random Forest follows.




## Variable Importance

① Multiple Logistic Regression: The coefficients of all the variables except "age" and "length" are significant, indicating that the two variables have no significant influence on default probability.

② Decison Tree: The variable "percent (loan amount/income)" is the most important factor, and "interest rate" and "home ownership" follows.

③ Random Forest: The variable "interest rate" is the most important variable in terms of the decrease of accuracy, and the variable "percent (loan amount/income)" is the most inportant variable with respect to the decrease of gini impurity.

# PART 4
# Model Combination

# 4.1 Bagging

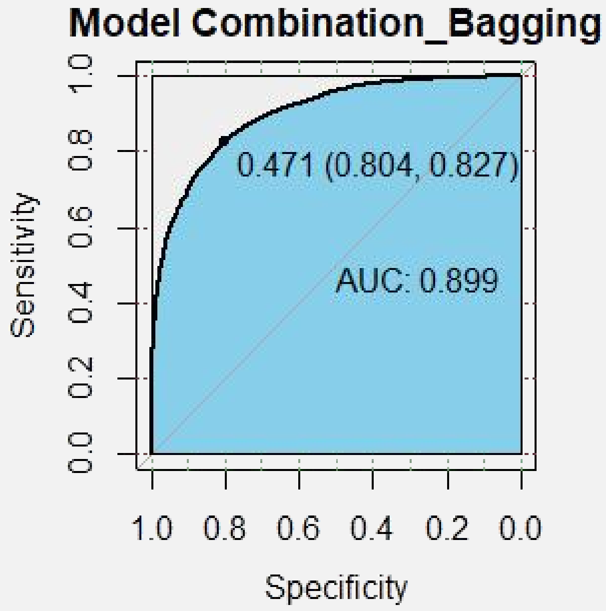
Bagging is a method to build a base classifier on each self-help training sample set and get the final category of test samples by voting.

Inspired by bagging algorithm, we select Logistic Regression, KNN and Decision Tree as base learners and use bagging to solve the classification problem. For the regression problem, the result is the mean of the base learner, and for the classification problem, the result is the probability or mean of each category derived from the percentage of the different categories.

The mean square error of the bagging model is 0.13.

The ROC curve on testing data is shown on the right and AUC=0.899.

We take the result which is larger than 0.5 as 1, which is less than 0.5 as 0. The sensitivity, specificity, and accuracy are 0.83, 0.80, 0.81, respectively.

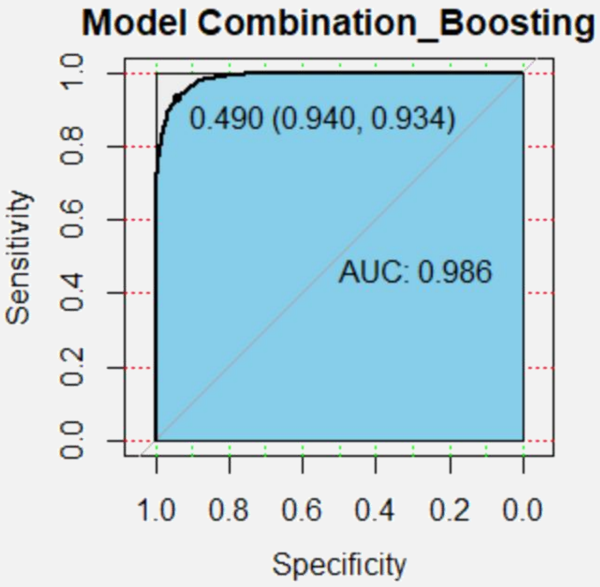


# 4.2 Boosting

Boosting is an integrated learning algorithm that builds multiple weak classifiers to predict the dataset, and then integrates the results with some strategy as the final prediction result.

Unlike Bagging, there is a dependence between weak classifiers in Boosting.

Inspired by XGBoost, we use Logistic Regression, Random Forest, and KNN in turn, to fit the residuals of the previous step respectively. The final score of a sample is to add the three scores predicted by the three models together.
The mean square error of the Boosting model is 0.06.
The ROC curve on testing data is shown on the right and AUC=0.986.
We take the result which is larger than 0.5 as 1, which is less than 0.5 as 0. The sensitivity, specificity, and accuracy are 0.93, 0.94, 0.93, respectively.



# PART 5
# Further Discussion

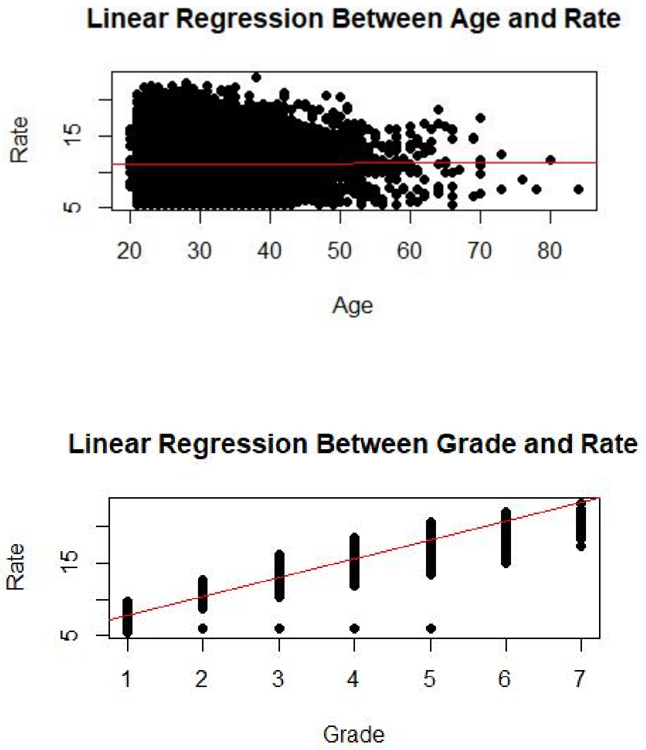
# 5.1 Loan Rate Analysis

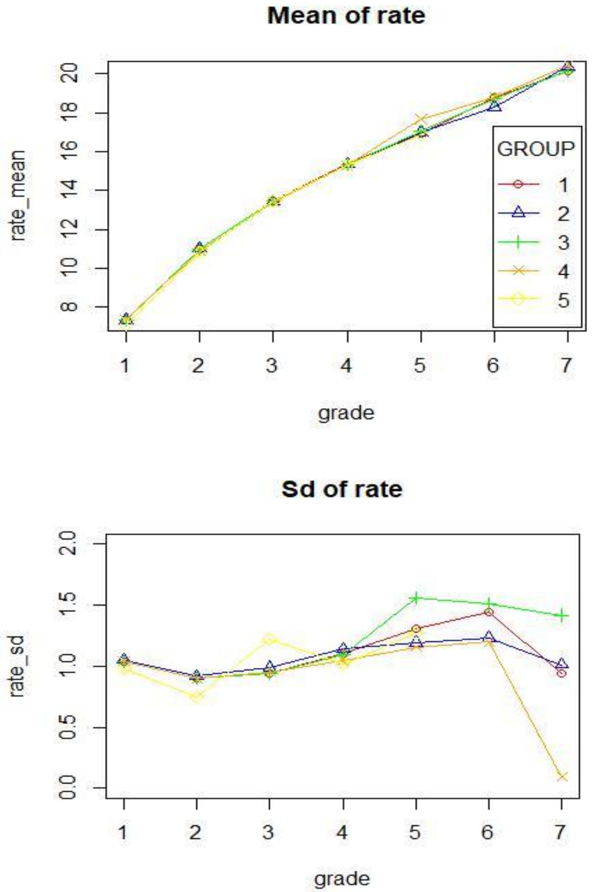
Rate and age don't have linear relationship.
There is a strong linear relationship between grade and rate:

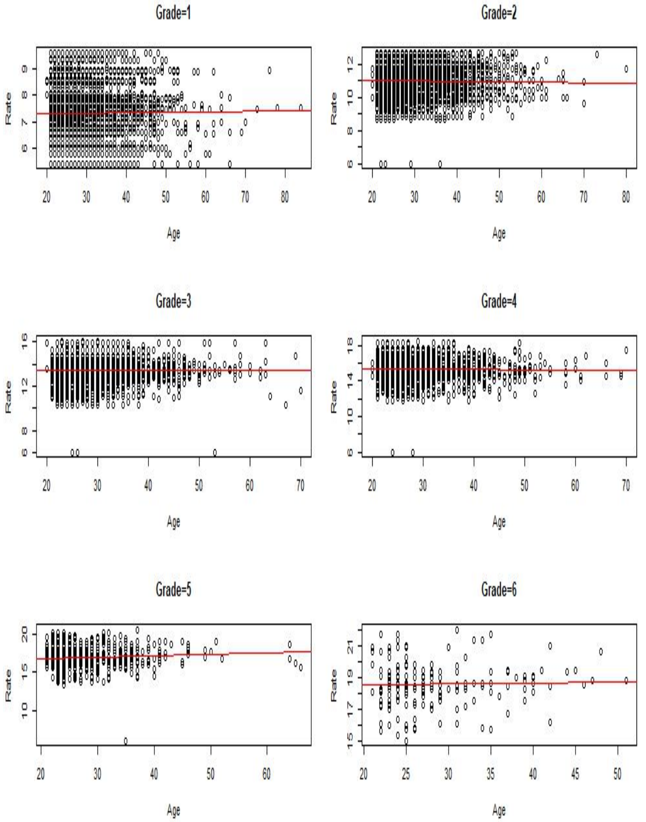
$$rate = 2.575 \times grade + 5.303$$

When people are in the same credit grade, the loan rate mean almost doesn't depend on their age stage/period (the mean is almost the same for different groups in the same grade.)

The R-squared is far less than 0.001 in the former six grades, which shows that almost no linear relationship exists.



# PART 6
# Conclusion

# 6.1 Conclusion

Model choices and comparison are two important ways for us to test the accuracy of the default risk prediction.

Five models:
Multiple logistic regression, K-Nearest-Neighborhood (KNN), Support Vector Machine (SVM), "decision tree" and Random Forest.

Three aspects to make the comparison:
sensitivity, specificity, and accuracy.
According to the confusion matrix, KNN has the highest sensitivity and Random Forest has the highest of both specificity and accuracy.

Suggestion:
The grade in the original data set is far from accurate, which means the grade systems of this company may be problematic. Therefore, based on the research result, we suggest the company should collect more background information of their borrowers for an all-around understanding or select a more suitable model then do the classification -- under the models we run, Random Forest is the most suitable one.

# THANKS FOR YOUR WATCHING

Simin Yu, Yuan Tan, Yuying Zhang, Kexin Sheng

**ACFBP 2021**