Business Analytics Final Report - Debt Default Classification

21500268, 박성찬 (Park Seongchan)

1. Executive summary

This project is a final task in the Business Analytics (MEC40091) class of Professor Oh Joo-hee of the Department of Management & Economics in the second semester of 2021. Recently, interest in data has increased and various methods have been introduced using it. In the business perspective, there are attempts to generate profits and reduce costs by using these methodology. Data analysis and prediction modeling from a business perspective should be followed by explanation of the results to some extent beyond simply improving performance, and clear goal setting is very important. In this respect, this report clearly defines the purpose of this project and first describes the motivation for choosing a project related to it. And through this project, I will talk about the value that business, organization, institution, class, group, and country can get.

The goal of this project is to create a model that can predict individual creditworthiness within the FinTech field that was usually interested. This allows companies to take preliminary action against people who are potentially incapable of repaying, and at the same time, individuals can also identify their credit status. This will ultimately help curb potentially occurring problems through an increase in understanding of each household's ability to repay debt from a social perspective. In addition, it is another value that this project can bring to the general characteristics of new customers by clustering data of various customers.

To do this, I downloaded data on personal financial status of about 150,000 people from Kaggle, a data analysis and prediction model competition platform, and created a predictable model through Logistic regression analysis. And I can use the new test data to confirm the performance of the model. Moreover, through the given data of 150,000 people, I could make 5 clusters through unsupervised learning through kmeans algorithm and identify new customer types through knn algorithm. Finally, this project will be able to clearly understand the analysis goals from the business analytics perspective, and to confirm the processes such as data preprocessing, modeling, and final performance verification to perform them.

2. Background

The field of finance is one of the parts that are very closely connected to our lives. In recent years, we have faced new services that we cannot experience in the past, beyond simply transferring money and handling financial affairs in untact (Non-face-to-face). In the past, if a very small number of people paid expensive fees and received personal asset management services (Private Banker services), Fintech could now create a model that could manage them and customize them according to individual situations. Generally, these innovative services are referred to as **FinTech**, which is a combination of finance and technology, and these two fields are combined and recently, data is being managed and utilized in various fields and are developing faster.

In this project, I will carry out a result that can predict the probability of default based on data on individual financial status in this field of Fintech. I am interested in solving problems in our society through technology while majoring in Management and ICT, and finance is one of the areas of interest because it is very closely connected to our lives. In fact, in order to understand this field a little more, I am interning at a company called 'Korea Financial Solution' this semester. One of the various services the company is doing is recommending loan products to customers, and I would like to carry out a project to predict defaults in connection with this service.

Part 1

Background _ Self introduction

- I am majoring in ICT convergence and management.
- I am interested in solving many problems in our daily lives with technology based on insights through data. 'Finance' is one of the areas of my greatest interest.
- I am interning this semester at this Fintech-related company.

한국금융솔루션

1. Loan comparison service
2. Deposit account recommendation service
3. Asset allocation service using 'Mydata' of users
4. Robo-advisor service

Part 1

Background _ Introduction of project (Motivation)

Interested in the financial problems closely related to our lives

1. I wanted to do a project that **deals with data in the financial sector** that I was interested in.
2. I wanted to do **Fintech-related projects** because I would continue to work in the field of Fintech after graduation.
3. I am interested in the subject of credit score prediction itself, so I can find out **how the personal data of customers affects credit score.**

Debt default means a state in which someone cannot pay interest on debt or repay the principal of debt. The reason why this default measurement is important is that if it is not done in time, unexpected problems may arise in the company's cash flow, and as a result, not only the company that made the loan but also the individual will suffer mental and physical damage through debt. So basically, loans are based on 'trust', so lenders should be able to accurately measure the likelihood of defaults for those who borrow money before lending (Searat Ali et al., 2018). Therefore, this project will develop a model that can evaluate repayment ability based on data on each individual's financial situation. This reduces the potential risk of lending institutions, and individuals can also check their credit ratings. It will also be able to coordinate appropriate interest rates and repayment periods based on this.

Part 1

Background _ Introduction of project

Importance of Default Risk (Searat Ali et al., 2018)

The debtor is unable to fulfill interest or principal repayment as set forth in the contract.

Reason	Result	Solution
Firm's future cash flow is not sufficient to cover interest payments.	Individual productivity may be lowered, mental stress, and suicide may occur.	It should maintain a better governance mechanism and reduce the asymmetry of information.

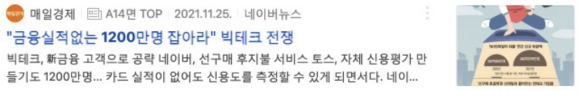
Source:
1. The Economic Terms Dictionary (Ministry of Strategy and Finance)
2. Searat Ali et al. (2018). Does corporate governance quality affect default risk? The role of growth opportunities and stock liquidity.

Part 3

Insights _ Impact and contribution

" My data service, which started in December in Korea "

Recommending products or services through own data
by checking **distributed data in various sector** at one.



Catch 12 million without financial performance:
Big Tech War

- Use other customers' financial **databases to create clusters** of type.
- Customers who do not have enough data based on clusters can **estimate financial status** by using only a small amount of data.
- It may not be **discriminated against by the service** regardless of the amount of data.

My data services in the financial sector, which started in Korea from the beginning of this month, have become a cornerstone of providing customized financial services based on information related to the finance of scattered individuals. It is expected that new value added can be created by crossing the field through various data, while another service is needed for customers who do not have enough financial performance (data). In fact, MZ generations and housewifes who have difficulty in lending due to lack of social activities or financial transactions may not be able to receive appropriate services. Therefore, I can group existing data and kmeans and knn algorithms to infer similar customer characteristics.

Part 1

Background _ Introduction of project (Goal)

Risk can be reduced by distinguishing people with repayment ability

1. A credit scores are evaluated in consideration of **financial conditions** such as consumers' income or debt (data).
2. The **loan rate or the limit can be presented** through the objectively evaluated credit scores.
3. The financial state of the client is grouped, and data is utilized for the users of the **similar group** and the information of the **new customer** can be inferred.

Finally, the ultimate goal of the project is: Use customers' financial data to identify people with sufficient repayment capabilities to minimize potential risks. This will also enable existing and new customers to comprehensively understand the information and situations of their financial status. To achieve this, I developed a model using data on the financial situation of a total of 150,000 people at Kaggle, a data analysis and prediction model competition platform.

3. Loading data

```
train <- read.csv('cs-training.csv')
```

```
str(train)
```

```
## 'data.frame':    150000 obs. of  11 variables:
##  $ DEFAULT               : int  1 0 0 0 0 0 0 0 0 0 ...
##  $ Age                   : int  45 40 38 30 49 74 57 39 27 57 ...
##  $ Delay_30_59_days      : int  2 0 1 0 1 0 0 0 0 0 ...
##  $ Delay_60_89_days      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Delay_90_days         : int  0 0 1 0 0 0 0 0 0 0 ...
##  $ Num_open_credit_loans : int  13 4 2 5 7 3 8 8 2 9 ...
##  $ Num_open_mortage_loans: int  6 0 0 0 1 1 3 0 0 4 ...
##  $ Credit_limit_on_debt  : num  0.766 0.957 0.658 0.234 0.907 ...
##  $ Debt_ratio            : num  0.803 0.1219 0.0851 0.036 0.0249 ...
##  $ Monthly_income        : int  9120 2600 3042 3300 63588 3500 NA 3500 NA 23684 ...
##  $ Num_dependents        : int  2 1 0 0 0 1 0 0 NA 2 ...
```

Variable	Meaning
DEFAULT (Target variable)	Experienced 90 days past due delinquency or worse
Age	Age of borrower in years
Delay_30_59_days	# of times borrower has been 30-59 days past due but no worse in the last 2 years
Delay_60_89_days	# of times borrower has been 60-89 days past due but no worse in the last 2 years
Delay_90_days	Times borrower has been 90 days or more past due
Num_open_credit_loans	# of Open loans (installment like car loan or mortgage) and Lines of credit (credit cards)
Num_open_mortage_loans	# of mortgage and real estate loans including home equity lines of credit
Credit_limit_on_debt	Total balance on credit cards and personal lines of credit except real estate and no installment debt like car loans divided by the sum of credit limits
Debt_ratio	Monthly debt payments, alimony,living costs divided by monthy gross income
Monthly_income	Monthly income
Num_dependents	# of dependents in family excluding themselves (spouse, children etc.)

Kaggle link - https://www.kaggle.com/c/GiveMeSomeCredit/overview

When a total of 10 variables (feature) are given to each user's finances after downloading data related to the debt fulfillment / default of a total of 150,000 people from the Kaggle dataset, try to create a binary classification model that can predict whether the customer will be able to repay the loan well within a given period of time. The shape of train data is (150000, 11) and the target variable is **DEFAULT**. The overall explanation of each variable can be confirmed through the above table, and the process of how the model was improved and how the model was improved is explained.

4. Basic data preprocessing

The basic preprocessing of data is performed. I will try to perform the process of how much NA value is, whether there is an Outlier value, and standardizing the overall distribution of the value if necessary.

```
# PART 1. Checking Outlier
total_df <- data.frame()
for (i in 2:ncol(train)){
  col_name <- colnames(train)[i]
  total_df <- rbind(total_df, c(col_name, summary(train[, i], digits = 7)))
}

colnames(total_df) <- c('Variable', 'Min', '1st_Qu', 'Median', 'Mean', '3rd_Qu', 'Max')
total_df
```

```
##                   Variable Min     1st_Qu    Median      Mean    3rd_Qu     Max
## 1                      Age   0         41        52  52.29521        63     109
## 2        Delay_30_59_days   0          0         0 0.4210333         0      98
## 3        Delay_60_89_days   0          0         0 0.2403867         0      98
## 4           Delay_90_days   0          0         0 0.2659733         0      98
## 5   Num_open_credit_loans   0          5         8   8.45276        11      58
## 6  Num_open_mortage_loans   0          0         1   1.01824         2      54
## 7    Credit_limit_on_debt   0 0.02986744 0.1541807  6.048438 0.5590462   50708
## 8              Debt_ratio   0  0.1750738 0.3665078  353.0051 0.8682538  329664
## 9          Monthly_income   0       3400      5400  6670.221      8249 3008750
## 10         Num_dependents   0          0         0 0.7572223         1      20
```

Firstly, check the statistics through a summary function for each variable to see if there are outliers. Overall, the results were confirmed to be good according to the characteristics of the data. However, there was some difficulty in objectively determining the value of the variable such as Credit_limit_on_debt, Debt_ratio, Monthly_income and so on because it has relative meaning. But the Age variable confirmed that the minimum value was zero, and I reprinted this data to see exactly what it meant to be zero.

```
train %>%
  dplyr::filter(Age < 20) %>%
  select(DEFAULT, Age, Debt_ratio, Monthly_income, Num_dependents)
```

```
##   DEFAULT Age Debt_ratio Monthly_income Num_dependents
## 1       0   0  0.4369272           6000              2
```

```
train <- train %>%
  dplyr::filter(Age >= 20)

summary(train$Age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    21.0    41.0    52.0    52.3    63.0   109.0
```

If I look at data that is under 20 years old, I can see one data, which is 0 years old and Monthly_income is 6000, so I can assume that it is the outlier. Therefore, this data is removed and train data is again built.

```
# PART 2. Checking NA
colSums(is.na(train))
```

```
##                 DEFAULT                     Age       Delay_30_59_days
##                       0                       0                       0
##       Delay_60_89_days          Delay_90_days  Num_open_credit_loans
##                       0                       0                       0
## Num_open_mortage_loans   Credit_limit_on_debt             Debt_ratio
##                       0                       0                       0
##         Monthly_income         Num_dependents
##                   29731                    3924
```

```
summary(train$Monthly_income)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##       0    3400    5400    6670    8249 3008750   29731
```

```
summary(train$Num_dependents)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   0.000   0.000   0.000   0.757   1.000  20.000    3924
```

Secondly, check the train dataset to see if there is NA for each variable. Checking the above results, I can confirm that there are some NA in Monthly_income and Num_dependents variables. Therefore, we will check the statistics again through the summary function for each variable. If you check the results, both variables have values from 0 to more positive values.

In the business goal of repaying loans, monthly_income is one of the most important variables. However, this value is NA, which is an omission of the customer's information, and I would like to build a model assuming that the user has no income at all in order to fill it. Also, the Num_dependents variable would have been worth if there was a family to support, but I thought it was safe to say that there was no NA value. Thus, both variables change NA to zero.

```
train <- train %>%
  mutate(Monthly_income = ifelse(is.na(Monthly_income), 0, Monthly_income),
         Num_dependents = ifelse(is.na(Num_dependents), 0, Num_dependents))

colSums(is.na(train))
```

```
##                 DEFAULT                     Age       Delay_30_59_days
##                       0                       0                       0
##       Delay_60_89_days          Delay_90_days  Num_open_credit_loans
##                       0                       0                       0
## Num_open_mortage_loans   Credit_limit_on_debt             Debt_ratio
##                       0                       0                       0
##         Monthly_income         Num_dependents
##                       0                       0
```

The results show that all variables do not have NA.

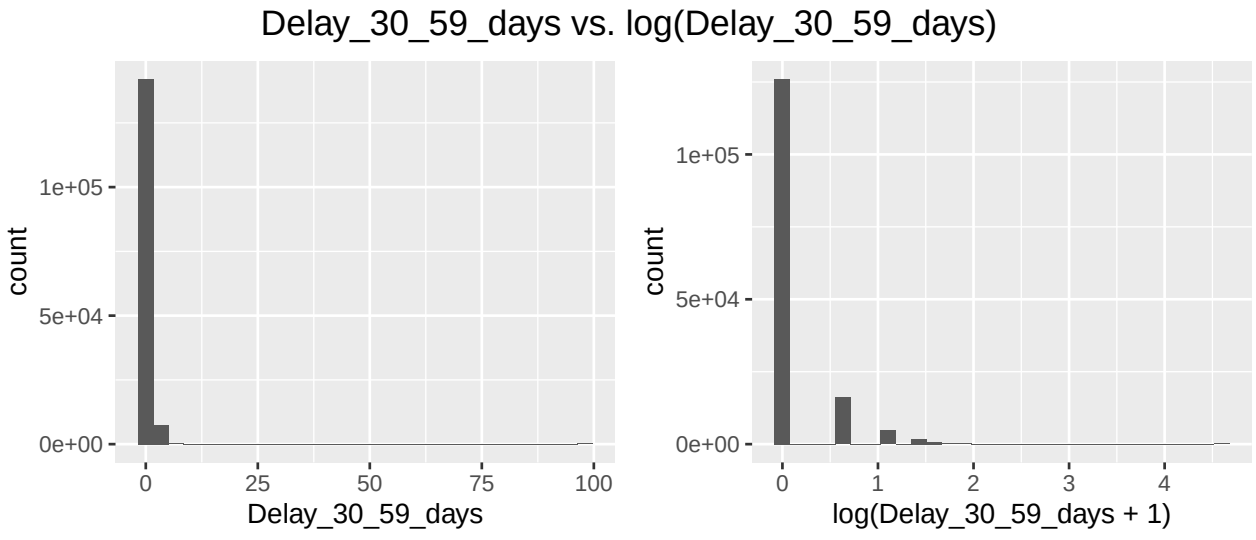
```
# PART 3. EDA, Standardizing
skewness(train)
```

```
##                 DEFAULT                     Age       Delay_30_59_days
##               3.4687734               0.1892388              22.5965873
##       Delay_60_89_days          Delay_90_days  Num_open_credit_loans
##              23.3311983              23.0868064               1.2152794
## Num_open_mortage_loans   Credit_limit_on_debt             Debt_ratio
##               3.4824420              97.6292964              95.1555976
##         Monthly_income         Num_dependents
##              119.9032853               1.6260549
```

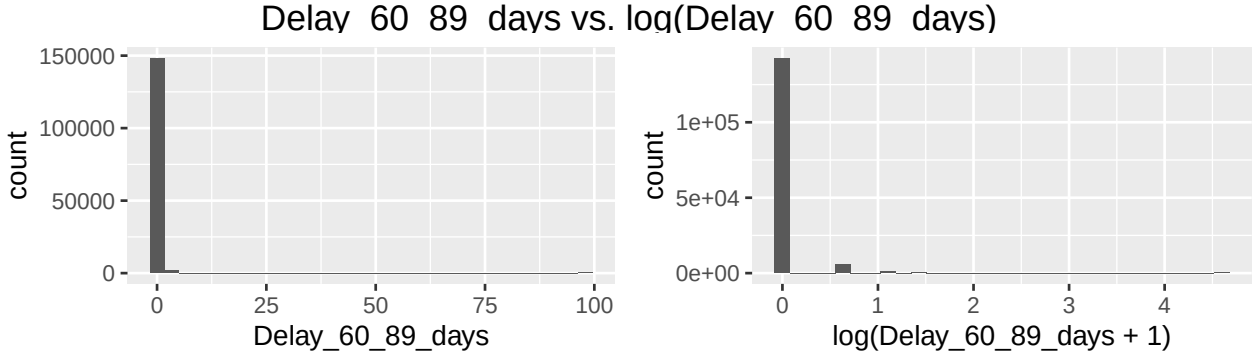
Next, I will try to identify the overall characteristics of the data. First, I check the skewness to check whether the distribution of each variable in the dataset follows the form of the normal distribution. Generally, it is said that the shape with a long tail to the right has a positive degree of skewness, which shows that the probability distribution is asymmetric and its shape is much different from the normal distribution. Therefore, to compensate for this, asymmetry is reduced through log function, and regularity is increased. The study found that the range of possible skewness, that is, the range of skewness that can be said to be relatively unbiased, ranges from -2 to 2 (George & Mallery, 2010).

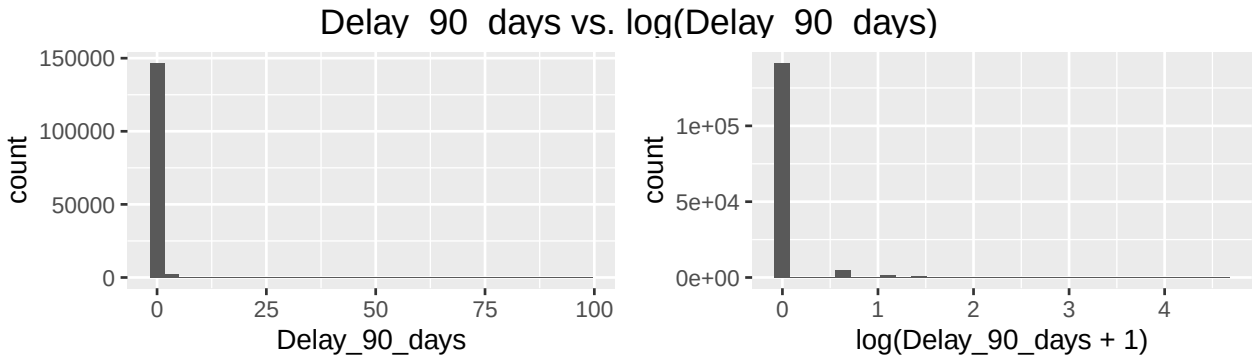
Calculating the degree of skewness for train dataset, I can confirm that all except for Age, Num_open_credit_loans, and Num_dependents variables are out of the ideal range. Therefore, all the remaining variables are standardized through log function.

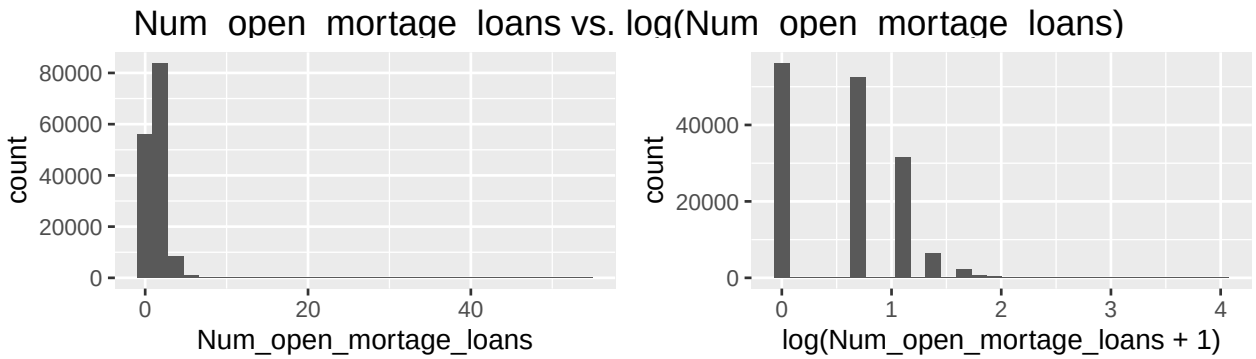
```
hist1 <- train %>% select(Delay_30_59_days) %>%
  ggplot(aes(x = Delay_30_59_days)) + geom_histogram()
hist2 <- train %>% select(Delay_30_59_days) %>%
  ggplot(aes(x = log(Delay_30_59_days + 1))) + geom_histogram()

title <- ggdraw() + draw_label('Delay_30_59_days vs. log(Delay_30_59_days)')
plot_grid(title, plot_grid(hist1, hist2), ncol = 1, rel_heights = c(0.1, 1))
```

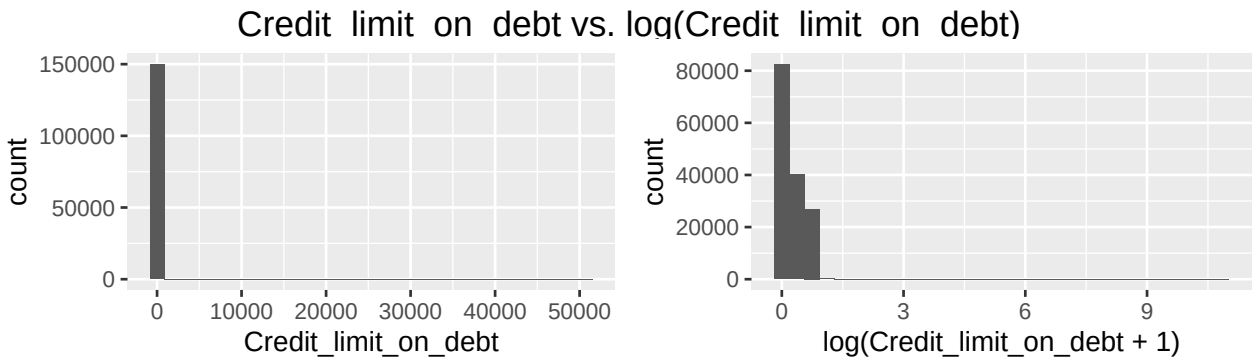


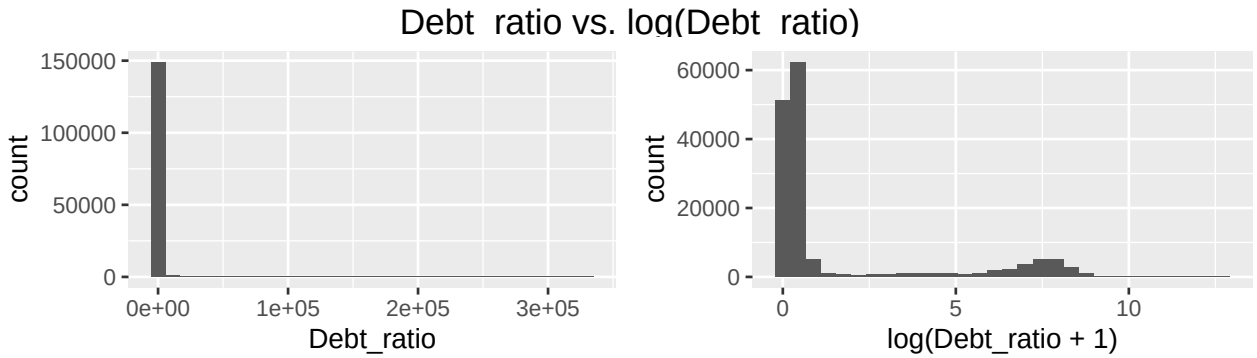
Before standardization, visualize the distribution of dataset before and after standardization, and let us directly check how the distribution changes before and after standardization through histogram.

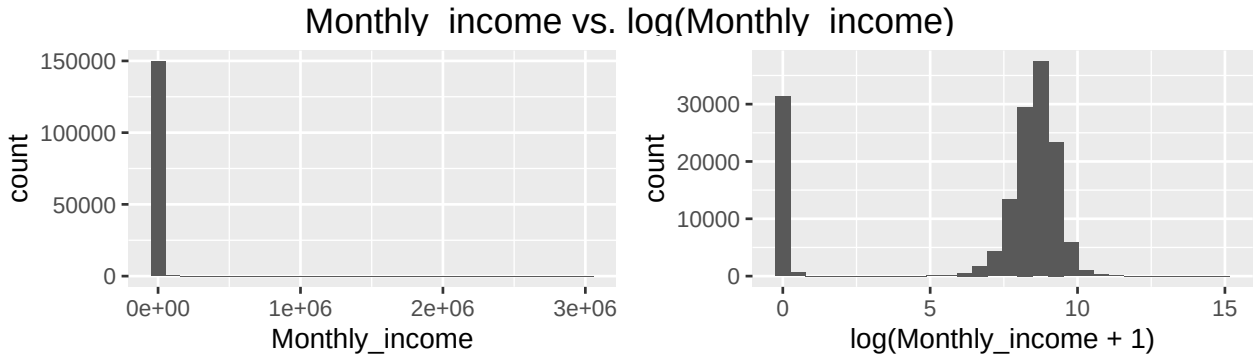

Delay_60_89_days vs. log(Delay_60_89_days)
count
Delay_60_89_days
log(Delay_60_89_days + 1)


Delay_90_days vs. log(Delay_90_days)
count
Delay_90_days
log(Delay_90_days + 1)


Num_open_mortage_loans vs. log(Num_open_mortage_loans)
count
Num_open_mortage_loans
log(Num_open_mortage_loans + 1)


Credit_limit_on_debt vs. log(Credit_limit_on_debt)
count
Credit_limit_on_debt
log(Credit_limit_on_debt + 1)





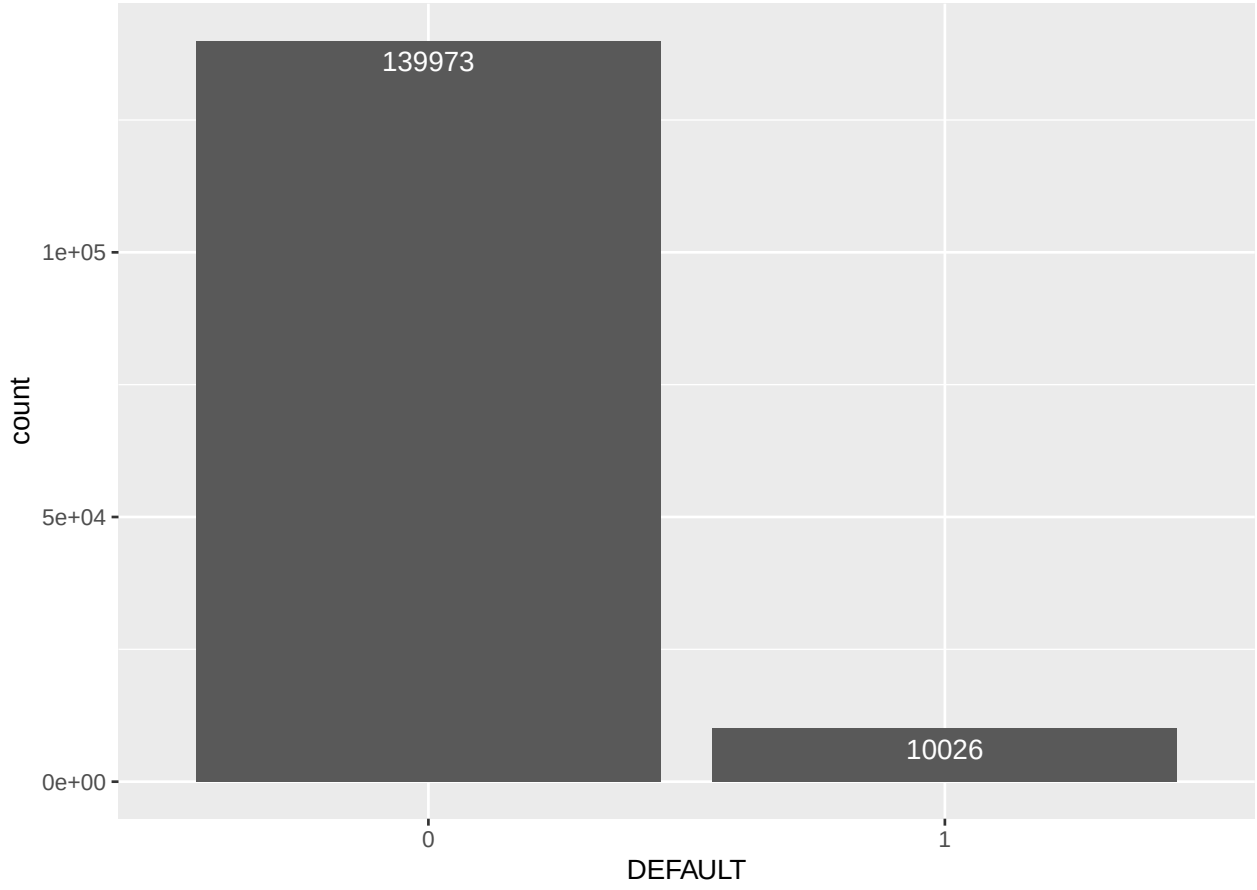

```
train <- train %>%
  mutate(Delay_30_59_days = log(Delay_30_59_days+1), Delay_30_59_days = 1/Delay_30_59_days,
         Delay_60_89_days = log(Delay_60_89_days+1), Delay_60_89_days = 1/Delay_60_89_days,
         Delay_90_days = log(Delay_90_days+1), Delay_90_days = 1/Delay_90_days,
         Num_open_credit_loans = log(Num_open_credit_loans + 1),
         Num_open_mortage_loans = log(Num_open_mortage_loans + 1),
         Credit_limit_on_debt = log(Credit_limit_on_debt + 1),
         Debt_ratio = log(Debt_ratio + 1), Monthly_income = log(Monthly_income + 1))
```

```
skewness(train)
```

```
##                DEFAULT                    Age       Delay_30_59_days
##              3.4687734              0.1892388              2.0949949
##       Delay_60_89_days         Delay_90_days  Num_open_credit_loans
##              4.3837731              4.3176369             -0.7330688
## Num_open_mortage_loans   Credit_limit_on_debt             Debt_ratio
##              0.2388435             11.7046881              1.7489676
##         Monthly_income         Num_dependents
##             -1.2917707              1.6260549
```

In the case of standardization through log, if the value is zero, the value diverges rather than the value, so add a very fine value to confirm the result. The final results show that most of the data are extremely biased, so there are some variables that seem to be visually different even if standardized through log, but nonetheless, the variables with the distribution of values are properly scattered (although not perfect) show the shape of the normal distribution. Also, when you actually convert the value and output the value again, you can see that there were quite a lot of values that were bigger than 10, but this time all of them disappeared.

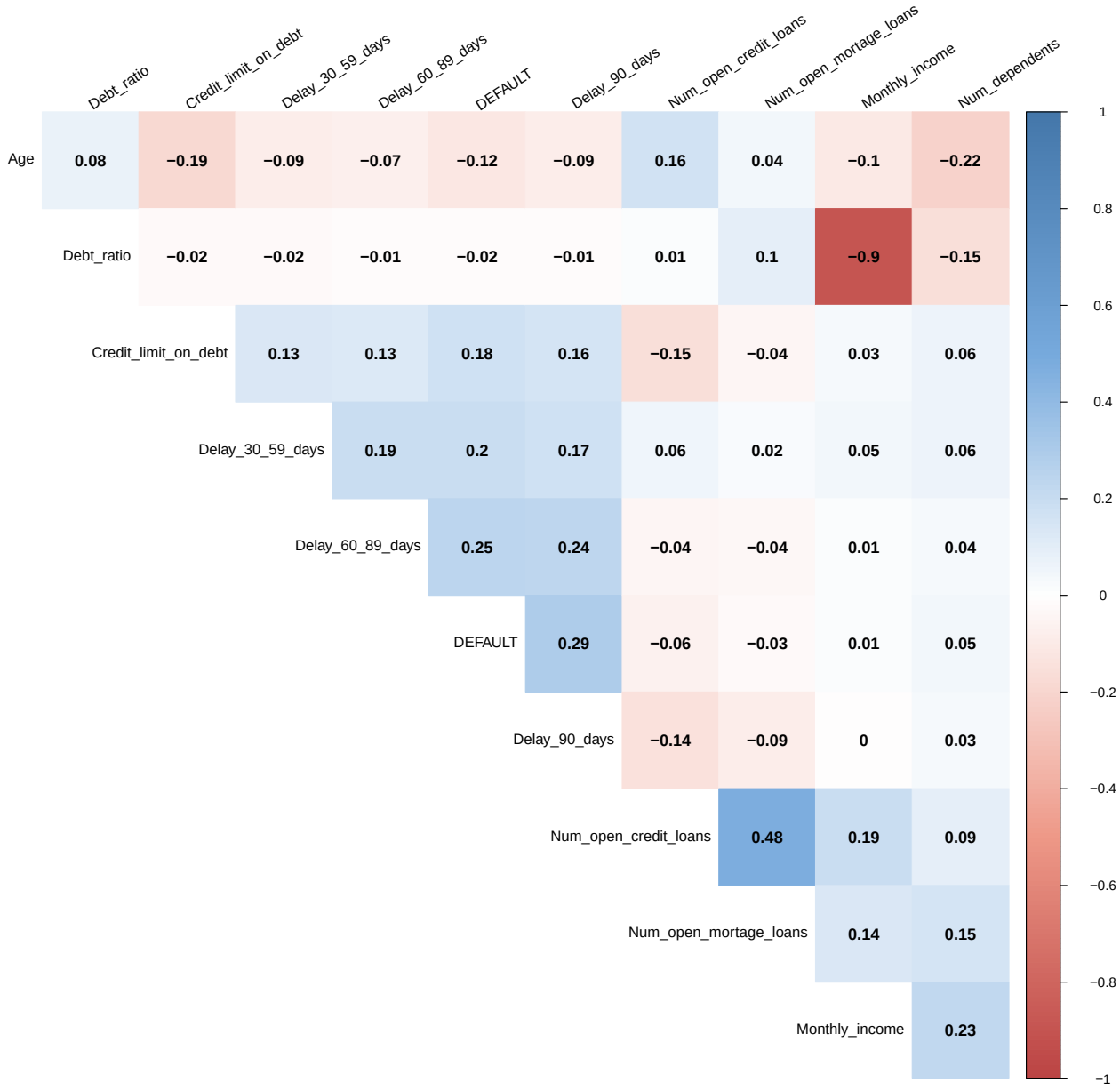
```
train %>%
  group_by(DEFAULT) %>% dplyr::summarise(count = n()) %>%
  ggplot(aes(x = reorder(DEFAULT, -count), y = count)) +
  geom_bar(stat = 'identity') + xlab('DEFAULT') +
  geom_text(aes(label = count), vjust = 1.5, colour = 'white')
```



This time, check out the target variable **DEFAULT** that I want to predict. This means that if you receive a loan, you will be repaid if you are 0, and if you are 1, you will not be able to repay it properly. The overall results show that the ratio between 0 and 1 is large, and Imbalanced Data is shown.

I try to increase understanding by checking the overall relationship of data through correlation coefficient. The correlation coefficients can be used to confirm the correlation between each of the two variables. The correlation between -1 and 1 is negative and positive. If the result is zero, you can say there's no relationship between the two variables. Then, check out the correlation coefficients for the train data.

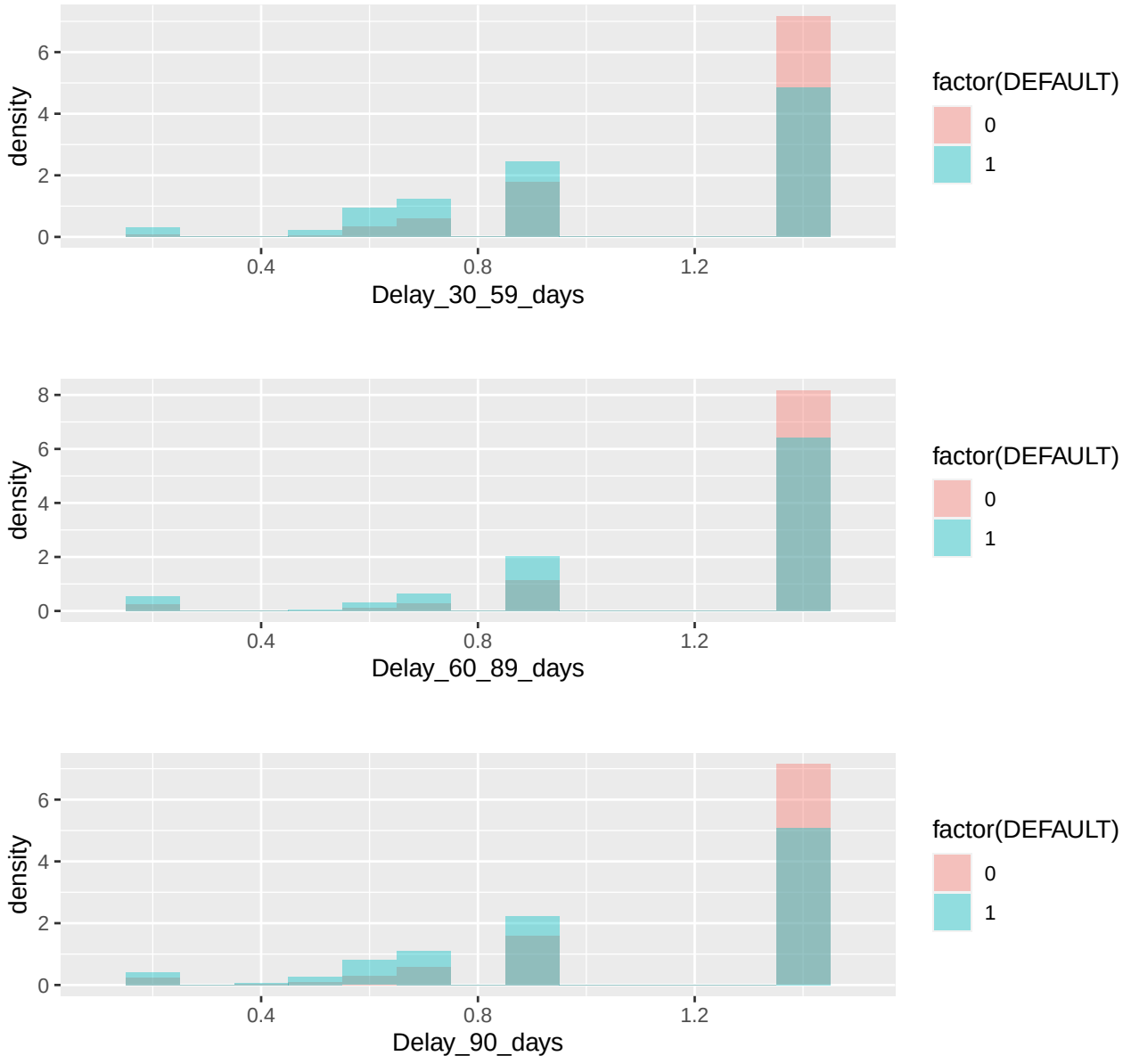
```
col <- colorRampPalette(c("#BB4444", "#EE9988", "#FFFFFF", "#77AADD", "#4477AA"))
corrplot(cor(train), method="color", col=col(200),
         type="upper", order="hclust", tl.cex = 0.9,
         addCoef.col = "black", tl.col="black", tl.srt=33,
         sig.level = 0.01, insig = "blank", diag=FALSE)
```

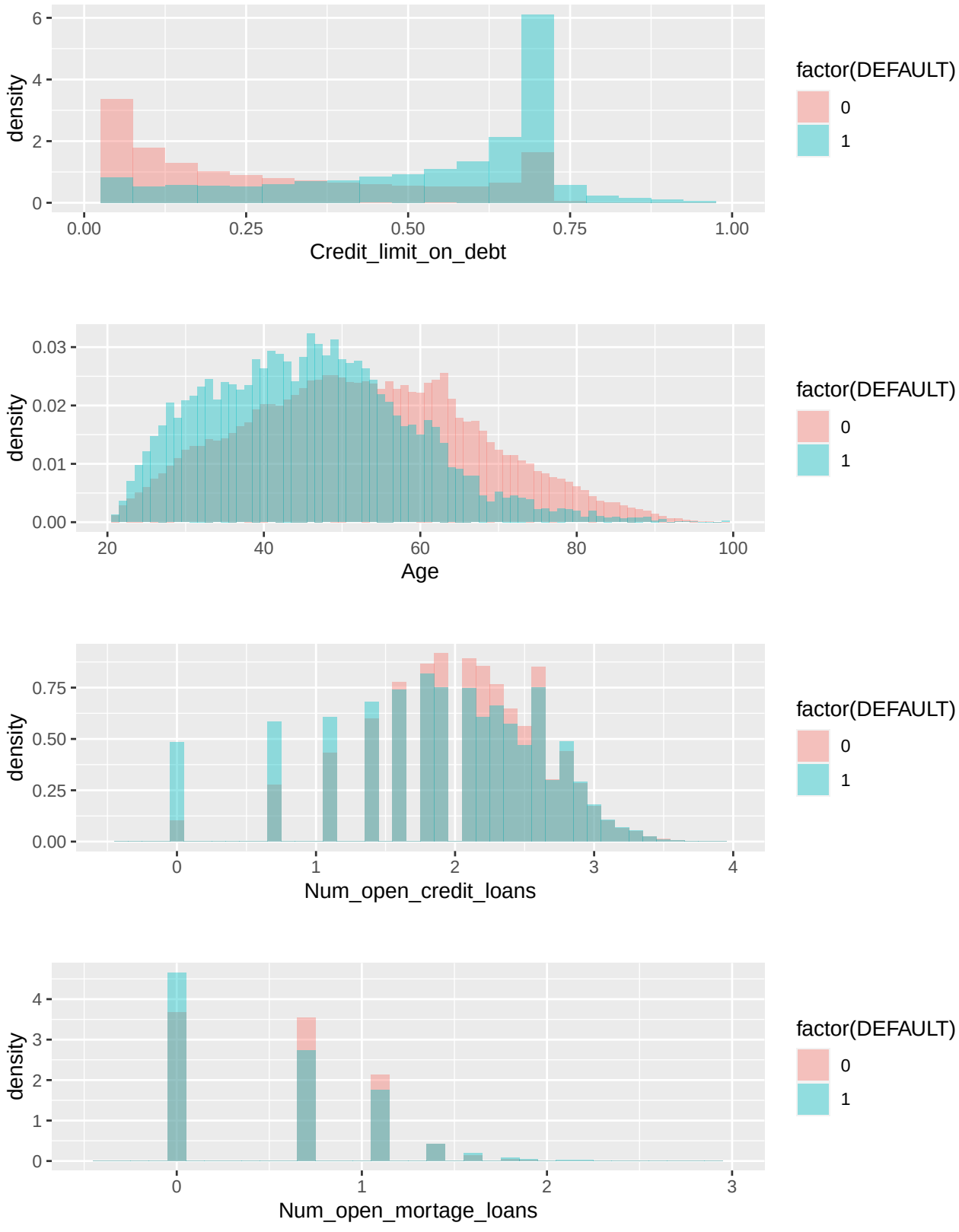


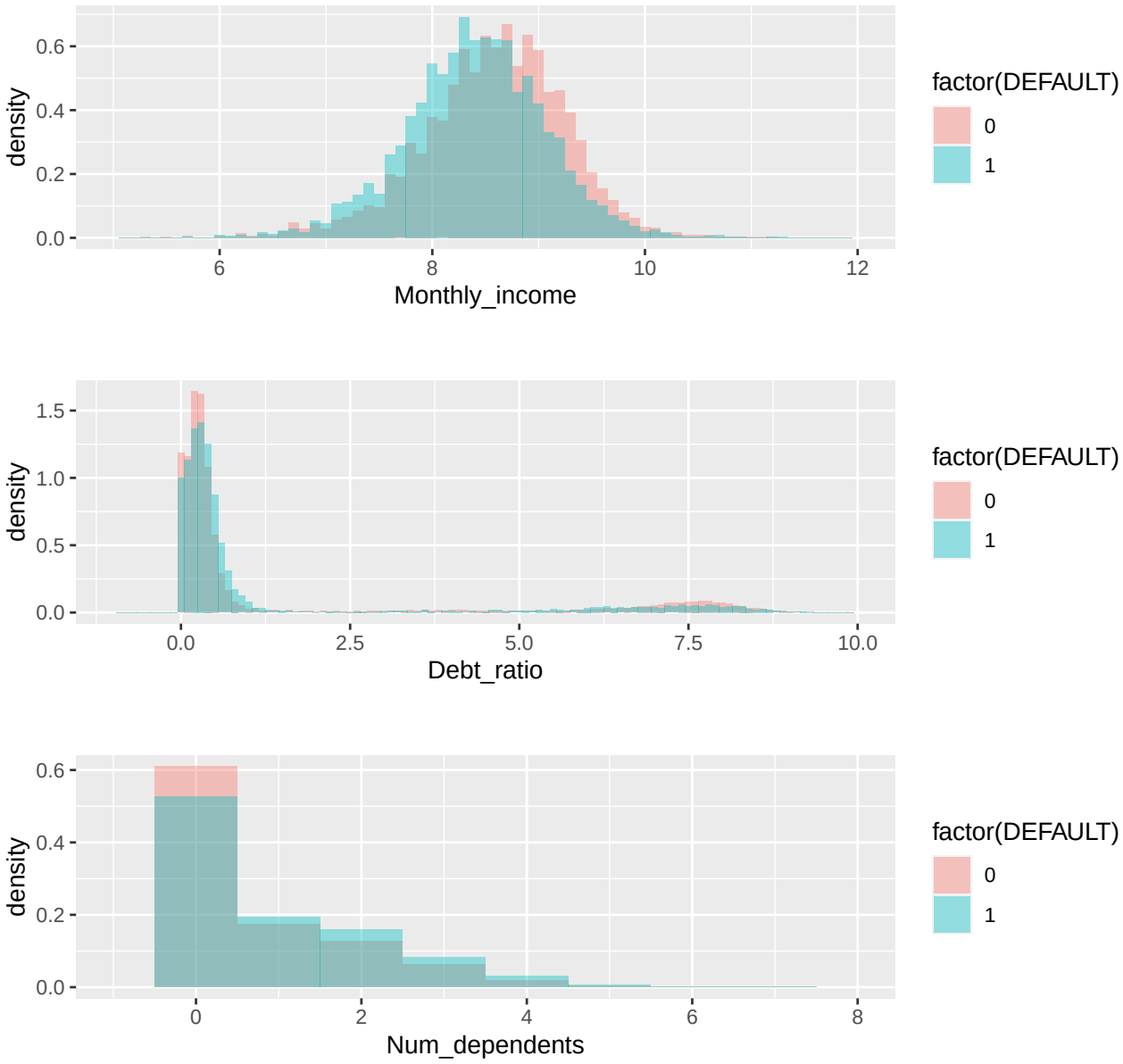
The result of the **correlation coefficient** for each variable can be confirmed as the above. It can be seen that the target variable, debt default, has a slight positive correlation with three variables (Delay 30 59 days, Delay 60 89 days, Delay 90 days) indicating the record of delinquency in the past. These three variables also have positive correlations among themselves. And the variable with high correlation coefficient and target variable was Credit limit on debt related to credit limit. On the other hand, the number of open credit loans, the number of mortage loans, the Monthly income, and the number of dependent families have a slight positive correlation between them.

If I interpret the meaning of the result of this correlation coefficient, past delinquency and credit limit data are closely related to default. Also, when monthly income was high, the number of open credit loan or mortage loan was high, and the number of dependents was high.On the other hand, monthly income and Debt ratio have a very strong negative correlation, so it can be confirmed that the proportion of debt is definitely low in case of high income.

```
train %>% ggplot() +
  geom_histogram(aes(x = Delay_30_59_days, y = ..density.., fill = factor(DEFAULT)),
                 binwidth = 0.1, position = 'identity', alpha = 0.4) +
  xlim(c(0.1, 1.5))
```




density
Credit_limit_on_debt
Age
Num_open_credit_loans
Num_open_mortage_loans
factor(DEFAULT)
0
1



Prior to the full-scale modeling work, the following visualization was made to examine the relationship with the target variables to predict all the data that have been preprocessed. Based on the target variable DEFAULT, I have drawn a graph showing the duration of the period that was not repaid on time, the credit rating, the age, the basic loan status, the degree of income and debt, and the density of the number of dependent families. The results of the previous correlation coefficients show that the data on the period data and the credit rating that were not repaid on time are somewhat correlated with the target variable. The results show that the repayment overdue period data is occurring slightly in the case of the failure to repay in general (1). However, it was also confirmed that some of the repayments were overdue (0). In the case of credit ratings, it can be seen that the blue result, which means that the overall repayment cannot be made, is spread to the right. In the age part, there were many defaults in younger people than expected, and there were many defaults when monthly profits were less than many.

5. Basic modeling

```
model_output <- function(model, train_df, target_variable, threshold){
  train_df <- train_df %>%
    mutate(pred_prob = predict(model, train_df, type = 'response'),
           pred = ifelse(pred_prob >= threshold, 1, 0))
  train_output <- table(pred = train_df$pred, actual = train_df[, target_variable])

  acc <- sum(diag(train_output)) / sum(train_output)
  F1 <- (train_output[1, 1] * 2) / ((train_output[1, 1] * 2) +
                                      train_output[1, 2] + train_output[2, 1])
  precision <- train_output[1, 1] / sum(train_output[1, ])
  recall <- train_output[1, 1] / sum(train_output[, 1])
  total_df <- data.frame(t(c(acc, F1, precision, recall, threshold)))
  colnames(total_df) <- c('Accuracy', 'F1', 'Precision', 'Recall', 'Threshold')

  return (total_df)}

logistic_model <- glm(DEFAULT ~ ., data = train, family = binomial(link = 'logit'))
summary(logistic_model)
```

```
##
## Call:
## glm(formula = DEFAULT ~ ., family = binomial(link = "logit"),
##     data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.9174  -0.3201  -0.2600  -0.2109   3.1079
##
## Coefficients:
##                           Estimate Std. Error z value Pr(>|z|)
## (Intercept)            -2.0174008  0.0806769 -25.006  < 2e-16 ***
## Age                    -0.0241950  0.0008899 -27.190  < 2e-16 ***
## Delay_30_59_days        0.7997217  0.0186375  42.909  < 2e-16 ***
## Delay_60_89_days        1.0231302  0.0231632  44.171  < 2e-16 ***
## Delay_90_days           1.3369928  0.0230601  57.979  < 2e-16 ***
## Num_open_credit_loans  -0.0799226  0.0222789  -3.587 0.000334 ***
## Num_open_mortage_loans  0.1237271  0.0279371   4.429 9.48e-06 ***
## Credit_limit_on_debt    0.5531399  0.0194673  28.414  < 2e-16 ***
## Debt_ratio             -0.0334170  0.0123602  -2.704 0.006859 **
## Monthly_income         -0.0214078  0.0094288  -2.270 0.023179 *
## Num_dependents          0.0542008  0.0097915   5.536 3.10e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 73616  on 149998  degrees of freedom
## Residual deviance: 59931  on 149988  degrees of freedom
## AIC: 59953
##
## Number of Fisher Scoring iterations: 6
```

```
for (cut_off in seq(0.5, 0.97, 0.01)){
  df <- model_output(logistic_model, train, 'DEFAULT', cut_off)
  output_df <- rbind(output_df, df)}

output1 <- train %>%
  mutate(pred_prob = predict(logistic_model, train, type = 'response'),
         pred = ifelse(pred_prob >= output_df[which.max(output_df$F1), 'Threshold'], 1, 0),
         correct = ifelse(DEFAULT == pred, 1, 0))
table(Actual = output1$DEFAULT, Pred = output1$pred)
```

```
##        Pred
## Actual      0      1
##      0 139971      2
##      1  10023      3
```

Since the preprocessing is completed, I use the data to perform basic modeling and check the results. First, I made a logistic model by inputting the entire data. The significant variables were examined again through the significance level, and as confirmed by the summary results of the model, Debt ratio and Monthly income were slightly lower and the significance level was high in the rest of the variables. Therefore, I should consider the method to utilize the whole variable a little more and perform additional modeling. I confirm the distribution of actual value and prediction value as the model made through this. The results show that when the actual prediction coincides with the actual prediction, both the actual and the prediction are zero, while the prediction value is 0 and the actual value is 1 and the prediction is much wrong. This means that the probability of predicting 1 Label that this model will not be able to repay the debt on time is very low, because the ratio difference between 1 and 0 is very severe in the data itself, as mentioned above, so I think that more Label learns more and more precisely about the case with 0. Therefore, to compensate for this, each target label distribution is adjusted.

```
both_sampling_df <- ovun.sample(DEFAULT ~ ., data = train, method = "both",
                                p = 0.5, N = nrow(train), seed = 1)$data
rose_df <- ROSE(DEFAULT ~ ., data = train, seed = 1)$data
rbind(original = table(train$DEFAULT),
      both_sampling = table(both_sampling_df$DEFAULT),
      rose = table(rose_df$DEFAULT))
```

```
##                    0     1
## original      139973 10026
## both_sampling  75015 74984
## rose           75015 74984
```

The methods to compensate for these unbalanced data are over-sampling, under-sampling, ROSE-sampling, and SMOTE. Among them, the method in **ovun.sample** is known to be a method of performing the combination of over and under sampling. It means a way to increase or reduce the overall data based on Imbalanced data. Also, the ROSE method proposed by Menardi and Torelli in 2013 is a Bootstrap-based technology that helps binary classification work in Imbalanced data. It is characterized by processing both continuous and categorical data by generating a comprehensive example (synthetic samples) from the conditional density estimates of two classes. Finally, if check the number of each data, I can see that the existing original data has little 1 and too much 0. On the other hand, both sampling and rose methods show that the ratio is properly distributed. I can create models through these two datasets, and check and compare the prediction performance.

```
both_model <- glm(DEFAULT ~ ., data = both_sampling_df, family = binomial(link = 'logit'))

for (cut_off in seq(0.5, 0.97, 0.01)){
  df <- model_output(both_model, train, 'DEFAULT', cut_off)
  both_output_df <- rbind(both_output_df, df)}

both_output1 <- train %>%
  mutate(pred_prob = predict(both_model, train, type = 'response'),
         pred = ifelse(pred_prob >= both_output_df[which.max(both_output_df$F1), 'Threshold'],
                       1, 0),
         correct = ifelse(DEFAULT == pred, 1, 0))
table(Actual = both_output1$DEFAULT, Pred = both_output1$pred)
```

```
##       Pred
## Actual      0      1
##      0 139010    963
##      1   9182    844
```

```
F1_Score(both_output1$DEFAULT, both_output1$pred)
```

```
## [1] 0.9647945
```

First, I create a model for a dataset with sampling as both method. I use the entire data to create a logistic model, and I can confirm the performance of the final model through the threshold value, which is the most optimal F1 score, through the ratio of actual and prediction values. When Imbalanced data was not treated separately, most values predicted zero. In this case, 0 was predicted much more, so 0 and 1 were not evenly distributed and predicted, but it was confirmed that the prediction frequency of 1 was higher than that of existing data. Next, I want to check the results of the data sampled by the ROSE method.

```
rose_model <- glm(DEFAULT ~ ., data = rose_df, family = binomial(link = 'logit'))

for (cut_off in seq(0.5, 0.97, 0.01)){
  df <- model_output(rose_model, train, 'DEFAULT', cut_off)
  rose_output_df <- rbind(rose_output_df, df)}

rose_output1 <- train %>%
  mutate(pred_prob = predict(rose_model, train, type = 'response'),
         pred = ifelse(pred_prob >= rose_output_df[which.max(rose_output_df$F1), 'Threshold'],
                       1, 0),
         correct = ifelse(DEFAULT == pred, 1, 0))
table(Actual = rose_output1$DEFAULT, Pred = rose_output1$pred)
```

```
##       Pred
## Actual      0      1
##      0 139259    714
##      1   9448    578
```

```
F1_Score(rose_output1$DEFAULT, rose_output1$pred)
```

```
## [1] 0.9647984
```

In this case, I changed the data extracted by the ROSE method under the same conditions as the previous examples to create a model. The results of the two methods are almost similar, if I calculate the Accuracy compared to the results of the previous both sampling. I will continue to revise and supplement the model by using the sampled data with the ROSE method, which had a slightly higher F1 score.

< A Previous Study on Modeling to Reflect Domain Knowledge >

1. Gross, Souleles (2002): The increase in credit card delinquency rate was found to be due to the credit recovery system rather than to the change caused by the individual risk attributes. Debtors are capable of actually paying back, but they want to increase borrowing due to the existence of a credit recovery system and strategically use the related system to increase the debtor's own benefits.
2. Kim, Y.J., Nam, J.H., Kim, S.B. (2013): Personal bankruptcy is likely to be an individual's strategic choice, not an inevitable choice due to an unfortunate accident.
3. Kim, H.B. (2014): The possibility of delinquency varies depending on the degree of influence of individual credit characteristics such as credit card factors and loan factors of ordinary financial users.
4. Ryu, J. Y., Jeon, J. G. (2017): In the analysis of the factors determining the delinquency rate for household loans, the risk of liquidity of borrowers such as loan interest rates, size, period and income is the main determinant of the delinquency rate.
5. Lee, S. Y (2015): The analysis of delinquency behavior of the debtor of the credit recovery committee proved that the lower the age, the higher the loan amount, the lower the credit rating, and the weaker the working conditions.
6. Park, J. S., Nam, J. H. (2017): In the analysis of the impact on the debtor's default risk, the higher the age, the higher the repayment amount compared to monthly income, the lower the reduction rate of total debt, and the longer the repayment period, the higher the risk of default.

Source link - Credit Recovery System by Financial Committee

I examined several previous studies related to default. Basically, most of the studies were used as statistical methodology, so there was a slight difference in the results of each study depending on the sample group. In fact, Lee Si-hyung (2015)'s study claimed that the lower the age, the more delinquency it is, but Park Jung-soo and Nam Ju-ha (2017)'s study, on the contrary, the higher the age, the more delinquency it is. Therefore, I understand that the results of the study vary according to the sample, but once again confirm that the data related to the age, income, working conditions, loan amount and interest rate, credit rating, etc. of the user mentioned in the previous study have an effect on the default of the debt.

Meanwhile, the credit recovery system is as follows in the study of Gross, Souleles (2002) and in the study of Kim Young-joon, Nam Ju-ha and Kim Sang-bong (2013). According to the Financial Supervisory Service data, 'Since various credit recovery support systems are being implemented for the financial debt delinquents who are suffering from the failure to repay their debts in time, it is very important to choose the appropriate method for their situation in order to return to normal economic activities as soon as possible.' In fact, there is a system such as supporting emergency living stabilization funds and student funds for overdue debts through personal workouts and micro finance support, and allowing debts to be repaid in installments. The above mentioned studies can be confirmed that these systems are used differently from existing purposes. So if possible, I want to check whether I can find the type of people who are capable of repaying debt but intentionally fail.

Based on the contents of these previous studies, I will make new variables that can help predict target variables, or add interaction terms in regression analysis to improve modeling a little more, and confirm and compare the results. I will use the existing data to further utilize the age, income, creditworthiness, and loan size of the customers mentioned in the previous study. This is basically to use the regression model to fully understand the characteristics and relationships of the data, and if these tasks are completed, I will check whether the performance can be improved by using models such as logistic regression.

6. Modeling

```
new_train <- read.csv('cs-training.csv')
new_train <- new_train[, -1]
```

```
new_train <- new_train %>%
  mutate(Retirement = ifelse(Age >= 60, 1, 0),
  Income = Monthly_income ** 2,
  Credibility1 = Delay_30_59_days**2 + Delay_60_89_days**2 + Delay_90_days**2,
  Credibility2 = Credit_limit_on_debt ** 2,
  Debt_size = Num_open_credit_loans**2 + Num_open_mortage_loans**2)
```

```
rose_df <- ROSE(DEFAULT ~ ., data = new_train, seed = 1)$data
logistic_model <- glm(DEFAULT ~ ., data = rose_df, family = binomial(link = 'logit'))

final_output_df <- data.frame()
for (cut_off in seq(0.5, 0.97, 0.01)){
  df <- model_output(logistic_model, new_train, 'DEFAULT', cut_off)
  final_output_df <- rbind(final_output_df, df)}

output <- new_train %>%
  mutate(pred_prob = predict(logistic_model, new_train, type = 'response'),
         pred = ifelse(pred_prob >= final_output_df[which.max(final_output_df$F1), 'Threshold'],
                       1, 0),
         correct = ifelse(DEFAULT == pred, 1, 0))

table(Actual = output$DEFAULT, Pred = output$pred)
```

```
##       Pred
## Actual      0      1
##      0 138972   1001
##      1   8889   1137
```

```
F1_Score(output$DEFAULT, output$pred)
```

```
## [1] 0.9656399
```

```
sum(diag(table(Actual = output$DEFAULT, Pred = output$pred))) / nrow(new_train)
```

```
## [1] 0.9340662
```

Based on the results of the previous studies, the derived variables related to age, income, creditworthiness, and loan size, which are considered to affect default, were created. For example, regarding age, I created a **Retirement variable** which means retirement based on the age of 60, which is the legal retirement age of Korea, and created a derivative variable through squared income or credit limit ratio so that it can make real income or credit more prominent. The results of this process show that the model has slightly increased the frequency and F1 score of predicting default compared to the previous results.

```
test <- read.csv('cs-test.csv')
```

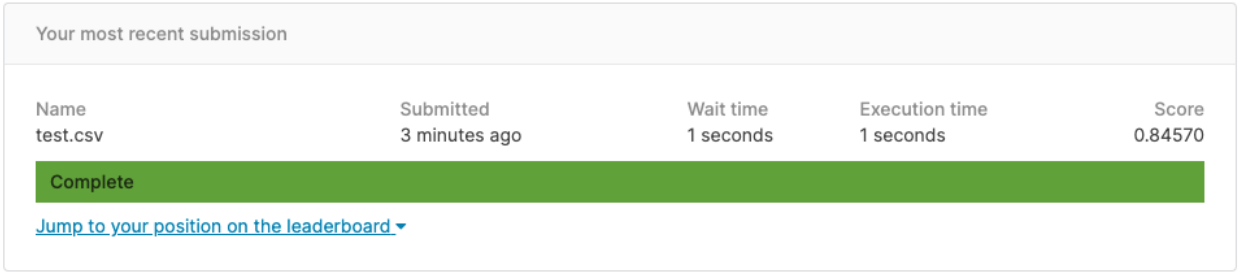
```
str(test)
```

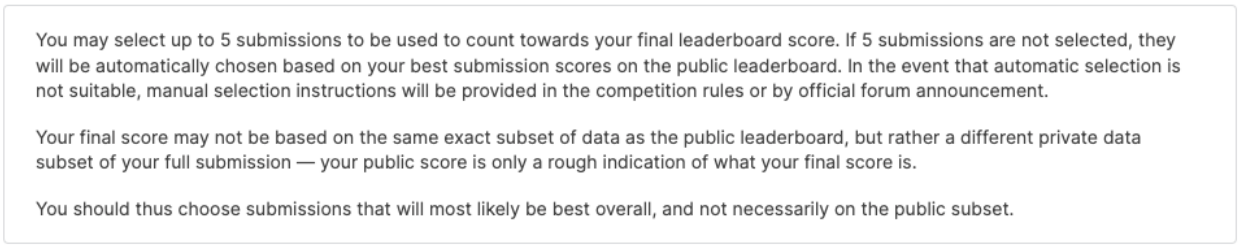
```
## 'data.frame':    101503 obs. of  11 variables:
##  $ DEFAULT               : logi  NA NA NA NA NA NA ...
##  $ Age                   : int  43 57 59 38 27 63 50 79 68 23 ...
##  $ Delay_30_59_days      : int  0 0 0 1 0 0 0 1 0 98 ...
##  $ Delay_60_89_days      : int  0 0 0 0 0 0 0 0 0 98 ...
##  $ Delay_90_days         : int  0 0 0 0 0 0 0 0 0 98 ...
##  $ Num_open_credit_loans : int  4 15 12 7 4 4 5 8 4 0 ...
##  $ Num_open_mortage_loans: int  0 4 1 2 0 0 0 1 1 0 ...
##  $ Credit_limit_on_debt  : num  0.8855 0.4633 0.0433 0.2803 1 ...
##  $ Debt_ratio            : num  0.1775 0.5272 0.6876 0.926 0.0199 ...
##  $ Monthly_income        : int  5700 9141 5083 3200 3865 4140 0 3301 NA 0 ...
##  $ Num_dependents        : int  0 2 2 0 1 1 3 1 0 0 ...
```

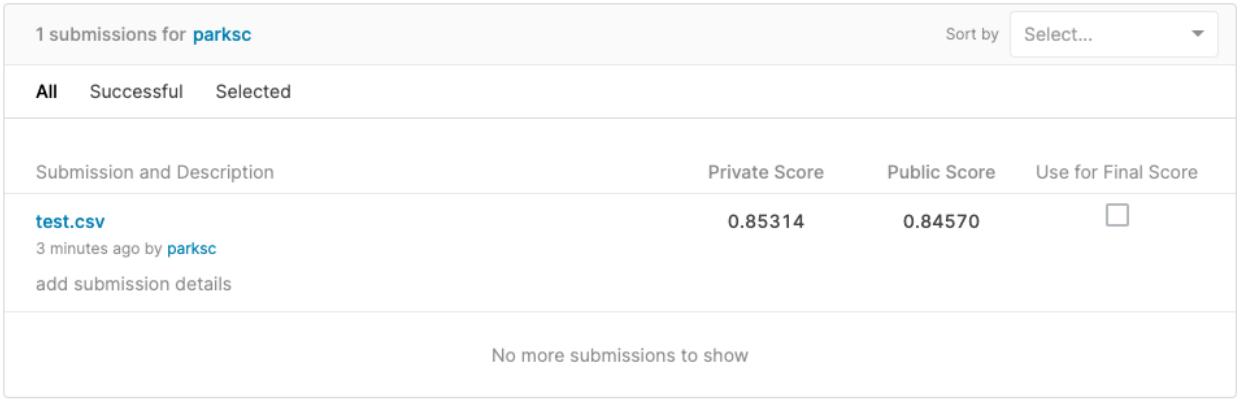
```
test <- test %>%
  mutate(Monthly_income = ifelse(is.na(Monthly_income), 0, Monthly_income),
         Num_dependents = ifelse(is.na(Num_dependents), 0, Num_dependents))

test <- test %>%
  mutate(Retirement = ifelse(Age >= 60, 1, 0),
         Income = Monthly_income ** 2,
         Credibility1 = Delay_30_59_days**2 + Delay_60_89_days**2 + Delay_90_days**2,
         Credibility2 = Credit_limit_on_debt ** 2,
         Debt_size = Num_open_credit_loans**2 + Num_open_mortage_loans**2)

test <- test %>%
  mutate(Delay_30_59_days = log(Delay_30_59_days+1), Delay_30_59_days = 1/Delay_30_59_days,
         Delay_60_89_days = log(Delay_60_89_days+1), Delay_60_89_days = 1/Delay_60_89_days,
         Delay_90_days = log(Delay_90_days+1), Delay_90_days = 1/Delay_90_days,
         Num_open_credit_loans = log(Num_open_credit_loans + 1),
         Num_open_mortage_loans = log(Num_open_mortage_loans + 1),
         Credit_limit_on_debt = log(Credit_limit_on_debt + 1),
         Debt_ratio = log(Debt_ratio + 1), Monthly_income = log(Monthly_income + 1),
         Income = log(Income + 1), Credibility1 = log(Credibility1 + 1),
         Credibility2 = log(Credibility2 + 1), Debt_size = log(Debt_size + 1))
```

```
test_output <- test %>%
  mutate(Id = 1:nrow(test),
         Probability = predict(logistic_model, test, type = 'response')) %>%
  select(Id, Probability)

head(test_output)
```

```
##   Id Probability
## 1  1   0.4611398
## 2  2   0.3702494
## 3  3   0.2310869
## 4  4   0.5971742
## 5  5   0.5486457
## 6  6   0.2539192
```





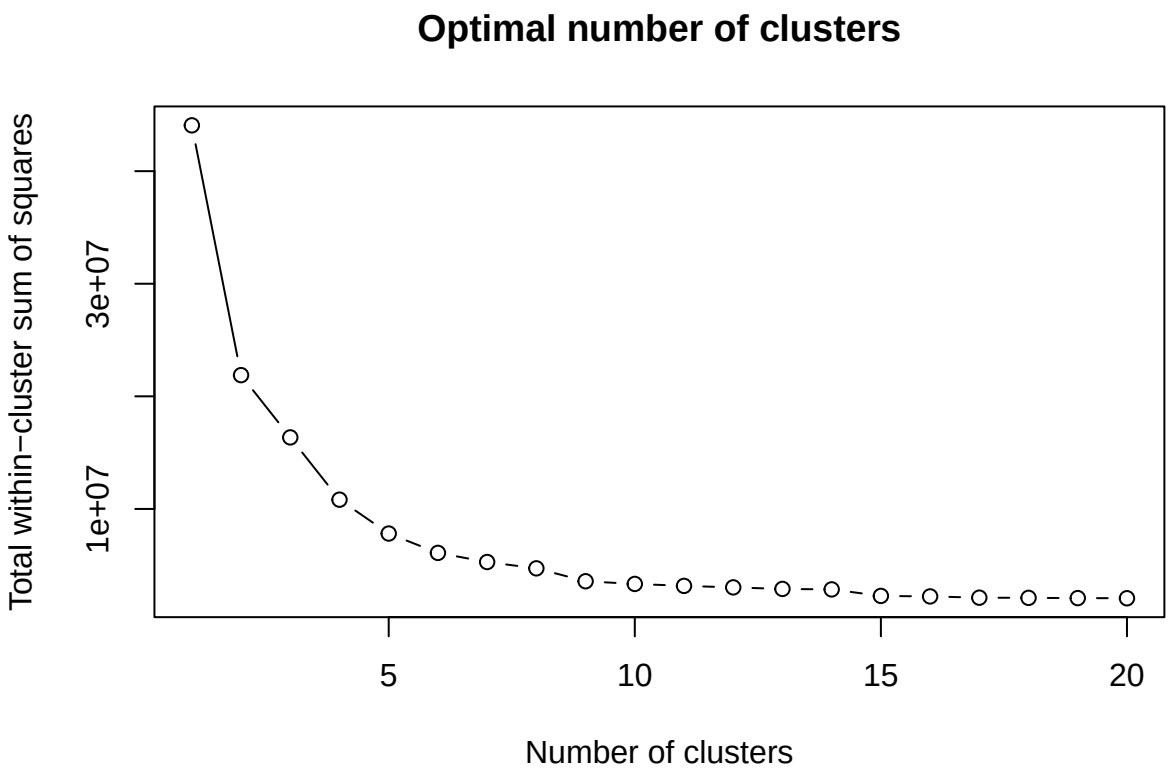


Let's check whether the Logistic model created above works well for new dataset. To this end, I will apply this model based on about 100,000 new test data that do not know whether the actual debt default. The basic preprocessing done in train data is performed equally so that the model can return the prediction value, and the probability value of default is returned from 0 to 1 through the model. Then, I make the probability value for this into a new data frame and csv file and submit it to the Kaggle site to check the result.

The results from the Kaggle homepage show that the score is about 0.8457, and the maximum value of the score of other people on the current basis is 0.86955. It may be difficult to say that performance is perfectly consistent, but it can still judge that the results from the simple Logistic regression model have some explaining power.

7. Insight

```
tot_withinss <- c()
for (i in 1:20){
  set.seed(1004)
  kmeans_cluster <- kmeans(curr_df, centers = i)
  tot_withinss[i] <- kmeans_cluster$tot.withinss
}
```



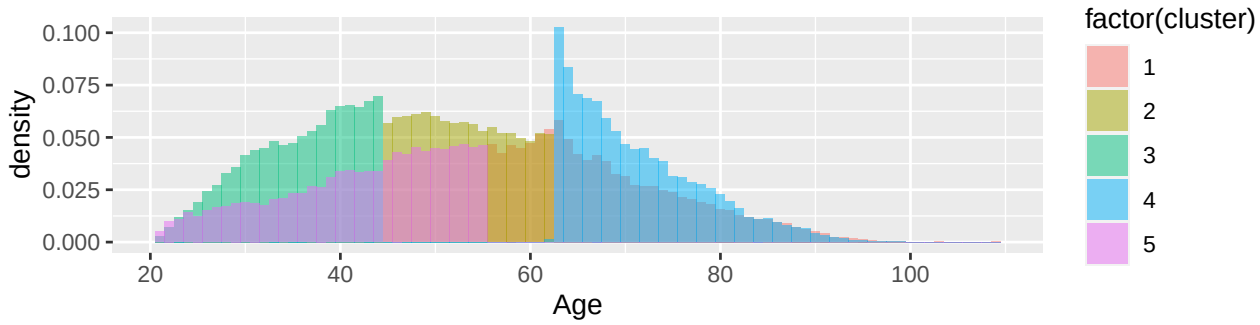
Next, as I talked at the beginning, I would like to extract some insights to have strengths in this service. I will use existing data to create clusters that match the characteristics of customers in each data, and check the information related to what potential characteristics of new customers are, and how many people in the cluster have not been able to repay on average. To do this, I try to cluster the first thing through **unsupervised learning through kmeans algorithm.** Data used for clustering is data except DEFAULT (Target variable). The kmeans algorithm has the characteristic that it can be used without detailed understanding of data. It has the advantage that it can confirm the result immediately if only the k value which means the number of clusters is input. However, it is difficult to find the appropriate k value, and the fact that the results vary a lot according to the k value is a limit. In the above process, 'elbow method' which is used to find the optimal k value in kmeans algorithm is used. This is to set a k value in the range where the WSS value, which means the variance of data in the cluster, is reduced.

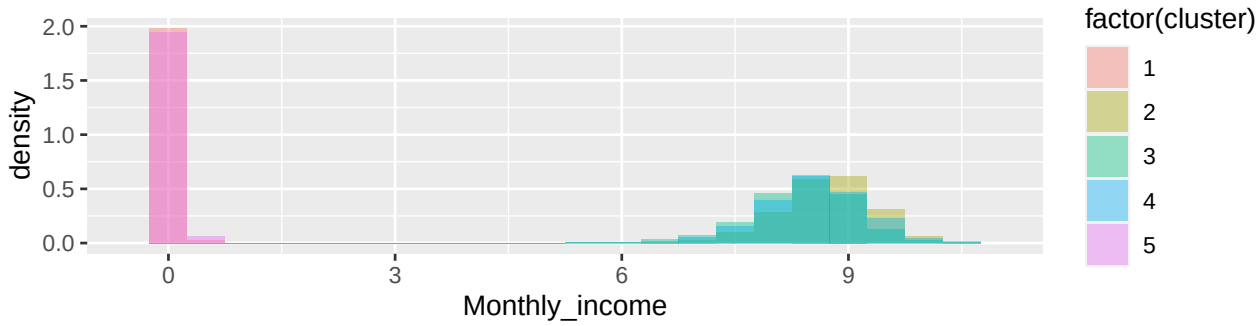
```
kmeans_5 <- kmeans(curr_df, centers = 5)
new_train$cluster <- kmeans_5$cluster
new_train$cluster <- as.factor(new_train$cluster)
table(new_train$cluster, new_train$DEFAULT)
```

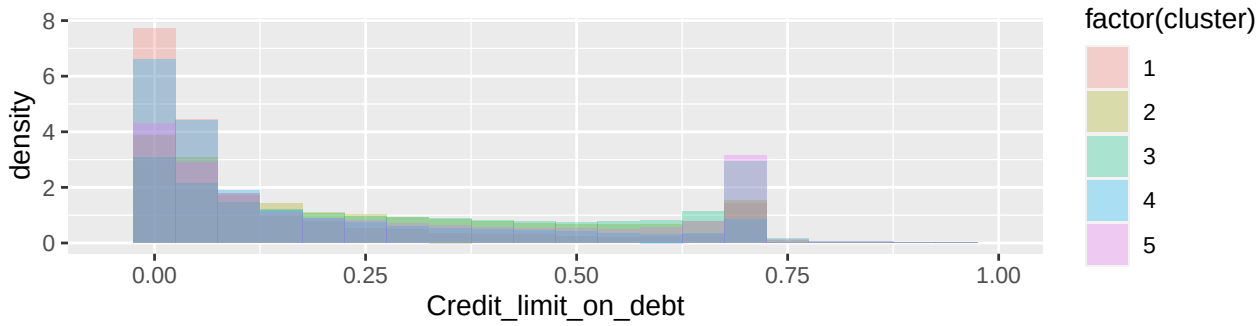
```
##
##         0     1
##   1 16291   416
##   2 47683  3542
##   3 35998  3925
##   4 26047   806
##   5 13954  1337
```

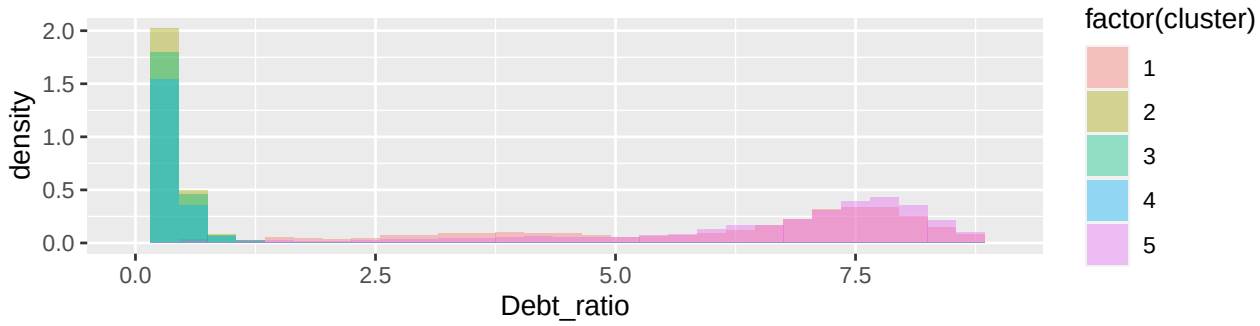
In the above results, I make the **final cluster in k = 5, which is the most optimal**, and check the characteristics of customers through total 5 clusters. And I check how the value of DEFAULT, a target variable, is distributed for each cluster. Then, try visualizing each cluster to see what characteristics each cluster has.

```
new_train %>% ggplot() +
  geom_histogram(aes(x = Age, y = ..density.., fill = factor(cluster)),
                 binwidth = 1, position = 'identity', alpha = 0.5)
```









```
aggregate(cbind(Age, Delay_30_59_days, Delay_60_89_days, Delay_90_days,
                Num_open_credit_loans, Num_open_mortage_loans, Credit_limit_on_debt,
                Debt_ratio, Monthly_income, Num_dependents, Income,
                Credibility1, Credibility2, Debt_size) ~ cluster, new_train, mean)
```

```
##   cluster      Age Delay_30_59_days Delay_60_89_days Delay_90_days
## 1       1 67.97666        0.1107716       0.03211413    0.03429425
## 2       2 53.17206        0.2242202       0.06945892    0.06553498
## 3       3 35.74571        0.2370877       0.08773797    0.09306439
## 4       4 71.10941        0.1356478       0.03419723    0.03251193
## 5       5 42.39612        0.1940328       0.07563632    0.08771737
##   Num_open_credit_loans Num_open_mortage_loans Credit_limit_on_debt Debt_ratio
## 1              1.990193              0.4781492            0.1638907  5.8055154
## 2              2.240881              0.7119770            0.2620495  0.3114604
## 3              1.957049              0.4831334            0.3385470  0.2839473
## 4              2.193764              0.5342647            0.1664939  0.2409490
## 5              1.843862              0.5218917            0.2966459  6.4104519
##   Monthly_income Num_dependents      Income Credibility1 Credibility2 Debt_size
## 1     0.00975319      0.1611899  0.01227132    0.1323124    0.1258411  3.715965
## 2     8.72354833      1.0130796 17.44667885    0.2898748    0.1884117  4.284766
## 3     8.39646016      1.0327631 16.79231066    0.3437869    0.2743313  3.651232
## 4     8.52847668      0.2810114 17.05641964    0.1472066    0.1091447  4.165232
## 5     0.02157784      0.4738081  0.02320142    0.3558805    0.2490137  3.416025
```

Through the visualization process, I will check the characteristics that are prominent in each cluster. I visualize the entire variable and print out only the visualization of the specially well-differentiated features. And the following general characteristics can be confirmed.

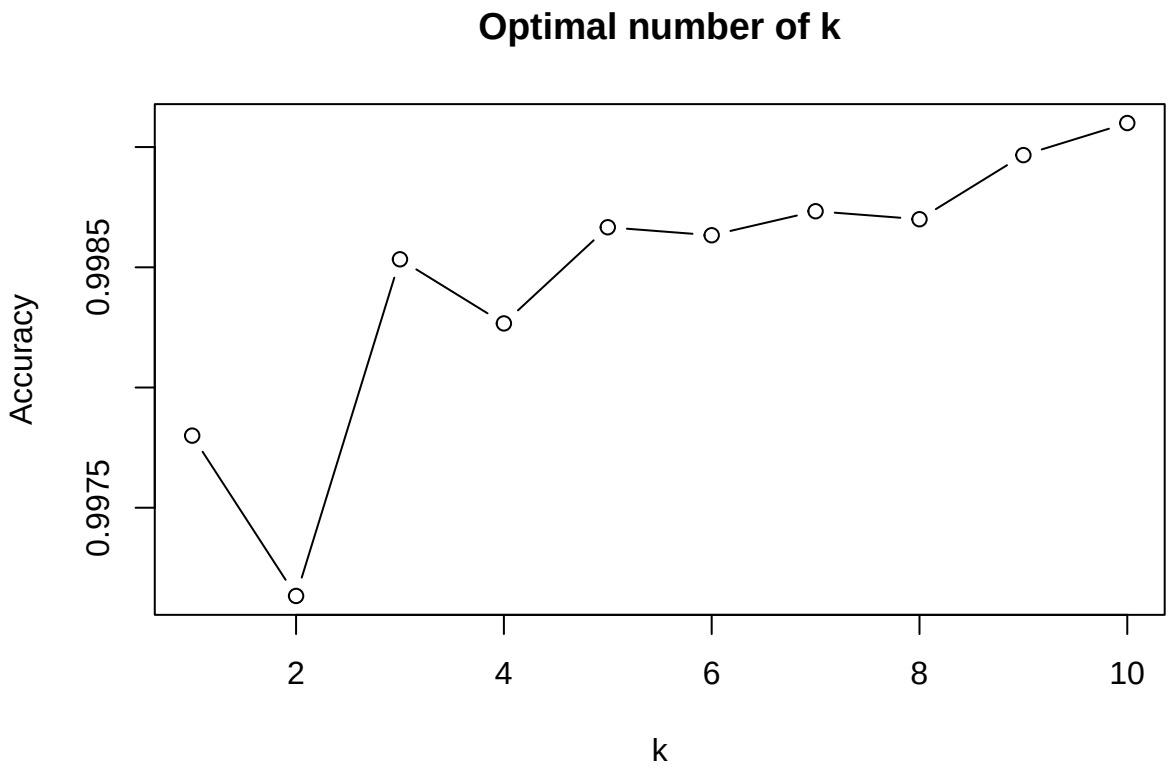
1. Cluster 1: Middle-aged (average 46.26 years), Little debt, High monthly income
2. Cluster 2: Young people (average 31.78 years), Have a little debt, Have a normal monthly income
3. Cluster 3: Middle-aged (average 54.94 years), Heavily in debt, With little monthly income
4. Cluster 4: Older people (average 61.08 years), Little debt, High monthly income
5. Cluster 5: High-aged people (average 77.85 years), Have a little debt, Have a normal monthly income

```
train_idx <- createDataPartition(new_train$cluster, p = 0.8, list = F)
train_data <- new_train[train_idx, ]
valid_data <- new_train[-train_idx, ]

rbind(table(train_data$cluster), table(valid_data$cluster))
```

```
##          1     2     3     4     5
## [1,] 13366 40980 31939 21483 12233
## [2,]  3341 10245  7984  5370  3058
```

In order to infer the types of new customers through these clusters, I try to create a **knn model that can predict clusters.** To achieve this, train and valid data are created at a rate of 80:20 through the CreateDataPartition function so that the number of clusters is evenly distributed. I also create a model through train data and check how well the clusters in the valid data can fit between the clusters predicted by the model.

```
vector_k <- c()
for (k_value in 1:10){
  knn_model <- knn(train = train_data[, c(-1, -17)], test = valid_data[, c(-1, -17)],
                   cl = train_data$cluster, k = k_value)
  vector_k[k_value] <- Accuracy(knn_model, valid_data$cluster)
}

plot(c(1:10), vector_k, type="b",
     main="Optimal number of k",
     xlab="k", ylab="Accuracy")
```



The knn algorithm is also a clustering algorithm based on whether there is as much data as k. In order to infer the optimal k, the accuracy value of the model is compared by changing k from 1 to 10. The current results show that when k is 10, the most accurate is, so make a model at 10.

```
data.frame(mean = colMeans(new_train[, c(-1, -12:-17)]))
```

```
##                                mean
## Age                     52.29555530
## Delay_30_59_days         0.19607531
## Delay_60_89_days         0.06448163
## Delay_90_days            0.06573192
## Num_open_credit_loans    2.08850857
## Num_open_mortage_loans   0.57383350
## Credit_limit_on_debt     0.25789699
## Debt_ratio               1.52518188
## Monthly_income           6.74393642
## Num_dependents           0.73740492
```

```
df <- t(data.frame(c(colMeans(new_train[, c(-1, -12:-17)]),
                     Retirement = 0, colMeans(new_train[, c(13, 14, 15, 16)]))))
rownames(df) <- NULL

knn(train = new_train[,c(-1, -17)], test = data.frame(df), cl = new_train$cluster, k=10)
```

```
## [1] 2
## Levels: 1 2 3 4 5
```

```
new_train[new_train$cluster == 2, ] %>% summarise(default_mean = mean(DEFAULT))
```

```
##   default_mean
## 1   0.06914592
```

As we can see from the previous results, the accuracy of cluster was the highest at k = 10. So, use the model to cluster arbitrary customer data and check the features that this customer will potentially have. Any customer data replaces all current data with an average value and clusters it to the corresponding value. If you check the results, you can see that the cluster of new customer data returned by the knn algorithm is number 2. Group 2 is a Young people (average 31.78 years), Have a little debt, Have a normal monthly income. In addition, the DEFAULT probability of these data can be confirmed to be about 0.069, so that the debt is almost not overdue.

8. Conclusion

So far, I have created a Logistic regression model that can predict the probability of default by using a total of 10 data on about 150,000 personal financial status provided by Kaggle. Generally, Data Science is called a region where computer science, math and statistics, and domain areas are all combined together. By checking the distribution while checking the statistics of the given data, I have confirmed what variables are basically related to default through previous studies. Input variables that can be used in the model through EDA processes such as remaking variables for customer age, income, creditworthiness, etc., or modifying existing variables are created. Then, the final model was built to confirm the results.

The current method is the Logistic model, which is the most basic model, and there will be other artificial intelligence (machine learning and deep learning) techniques that can improve performance. The logistic regression model is a model specialized in binary classification problems and returns the probability value belonging to one of the two labels using the logit function. Because it is a probability value, the range of the value is 0 to 1, and the commonly used cut-off value is 0.5, but the result is also confirmed by adjusting this value according to the form of the problem. In fact, while adjusting this value, you may raise the Precision or Recall value that is more suitable for the purpose of modeling. The advantage of the logistic regression model is the regression model, which is an efficient model that does not require a large amount of calculation resources. Also, it is easy to understand because it is easy to learn. However, logistic regression is limited to solve nonlinear problems, and performance itself is not superior to other machine learning algorithms.

Other machine learning models are also called black box models because they have a limitation that they have less explaining power for each result. Therefore, although the performance may be relatively low, I have built a final model by confirming the influence of each variable on the target variable and the explanatory power based on the traditional regression model. The final results were also confirmed that there was no significant difference from the models of others who participated in the Kaggle competition, as can be seen above.

Finally, I created a model that can predict defaults based on customer financial data, as I mentioned earlier in this project. In addition, for customers who do not have enough data, I have searched for similar types of customers through cluster analysis and reasoned for the characteristics of the customers. Through these processes, it is possible to improve the understanding of the credit status of individuals while receiving the recommendation of customized services to individuals. It will also contribute to reducing physical and mental harm, such as individual suicides caused by debt. From a social perspective, not only can the potential risk of the lender be reduced, but it will also be able to present appropriate interest rates and repayment periods to each customer.

Part 3

Insights _ Impact and contribution

< References >

1. Ali, S., Liu, B. & Su, J. (2018). Does corporate governance quality affect default risk? The role of growth opportunities and stock liquidity. International Review of Economics and Finance, 58 422-448.
2. David, B. G and Nicholas S. S. (2002). Do Liquidity Constraints and Interest Rates Matter for Consumer Behavior? Evidence from Credit Card Data. The Quarterly Journal of Economics Vol. 117, No. 1, pp. 149-185.
3. Kim, Y.J., Nam, J.H., Kim, S.B. (2013). Analysis of the Causes of the Increase in Personal Bankruptcy in Korea: Focused on the Post-2000 International Regional Research Volume 17 No.1, pp. 149-170.
4. Kim, H.B. (2014). The Effect of Credit Characteristics of the Public Financial Users on Credit Guarantee Accidents (Master Degrees).
5. Ryu, J. Y., Jeon, J. G. (2017). An Empirical Analysis of Determinants of Household Loan Products and Overdue Rate.Cap, pp. 363-390.
6. Park, J. S., Nam, J. H. (2017). An Analysis of the Effectiveness of Personal Workout Systems: Analysis of Korean Economy, Volume 23, pp. 1-53.
7. Lee, S. Y (2015). A Study on the Delinquencies of the Small Loan Debtors of the Credit Recovery Commission (Master Degrees).