MATH154 Team Challenge

Loading the packages:

```
library(e1071)
library(ggplot2)
library(plyr)
library(tidyverse)
```

EDA

We began our analysis by first loading the training data set and then examine the predicators.

```
data_train <- read.csv('data/cs-training.csv')
colnames(data_train)
```

```
##  [1] "X"
##  [2] "SeriousDlqin2yrs"
##  [3] "RevolvingUtilizationOfUnsecuredLines"
##  [4] "age"
##  [5] "NumberOfTime30.59DaysPastDueNotWorse"
##  [6] "DebtRatio"
##  [7] "MonthlyIncome"
##  [8] "NumberOfOpenCreditLinesAndLoans"
##  [9] "NumberOfTimes90DaysLate"
## [10] "NumberRealEstateLoansOrLines"
## [11] "NumberOfTime60.89DaysPastDueNotWorse"
## [12] "NumberOfDependents"
```

Portion of defaulted

```
mean(data_train$SeriousDlqin2yrs)
```

```
## [1] 0.06684
```

We then check each feature with **summary()** and see which of these features have null data and how many.

```
data_col <- colnames(data_train)
for(i in 2:12){
  print(data_col[i])
  print(summary(data_train[,i]))
}
```

```
## [1] "SeriousDlqin2yrs"
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.00000 0.00000 0.00000 0.06684 0.00000 1.00000
## [1] "RevolvingUtilizationOfUnsecuredLines"
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##     0.00     0.03     0.15     6.05     0.56 50708.00
## [1] "age"
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.0    41.0    52.0    52.3    63.0   109.0
## [1] "NumberOfTime30.59DaysPastDueNotWorse"
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   0.000   0.000   0.421   0.000  98.000
```

```
## [1] "DebtRatio"
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##      0.0      0.2      0.4    353.0      0.9 329664.0
## [1] "MonthlyIncome"
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##       0    3400    5400    6670    8249 3008750   29731
## [1] "NumberOfOpenCreditLinesAndLoans"
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   5.000   8.000   8.453  11.000  58.000
## [1] "NumberOfTimes90DaysLate"
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   0.000   0.000   0.266   0.000  98.000
## [1] "NumberRealEstateLoansOrLines"
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   0.000   1.000   1.018   2.000  54.000
## [1] "NumberOfTime60.89DaysPastDueNotWorse"
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.0000  0.0000  0.2404  0.0000 98.0000
## [1] "NumberOfDependents"
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   0.000   0.000   0.000   0.757   1.000  20.000    3924
```

This suggests that only monthly income and number of dependents have missing data, which we would later either fill in or drop. We then examine each variable to check for the existence of outliers

Revolving-Utilization-Of-Unsecured-Lines

For the second variable **Revolving-Utilization-Of-Unsecured-Lines**, which measures the total balance on credit card divided by sum of credit limits (amounts owing divided by total available for borrowing), the max number is 50708. That number is unlikely as we can't borrow beyond the limit by that much. Probably outlier?

Take a look at Observations with **Revolving-Utilization-Of-Unsecured-Lines** > 1 and > 100. There are 3338 obs with **Revolving-Utilization-Of-Unsecured-Lines** > 1 and 223 obs with **Revolving-Utilization-Of-Unsecured-Lines** > 100, probably treat **Revolving-Utilization-Of-Unsecured-Lines** > 100 as outliers?

```
g1 <- subset(data_train,RevolvingUtilizationOfUnsecuredLines>=1)
g2 <- subset(data_train,RevolvingUtilizationOfUnsecuredLines>=100)
summary(g1)
```

```
##        X          SeriousDlqin2yrs RevolvingUtilizationOfUnsecuredLines
##  Min.   :   163   Min.   :0.0000   Min.   :    1.00
##  1st Qu.: 38548   1st Qu.:0.0000   1st Qu.:    1.02
##  Median : 76612   Median :0.0000   Median :    1.07
##  Mean   : 75818   Mean   :0.3718   Mean   :  258.46
##  3rd Qu.:112457   3rd Qu.:1.0000   3rd Qu.:    1.30
##  Max.   :149974   Max.   :1.0000   Max.   :50708.00
##
##       age        NumberOfTime30.59DaysPastDueNotWorse   DebtRatio
##  Min.   :21.00   Min.   : 0.000                       Min.   :    0.001
##  1st Qu.:34.00   1st Qu.: 0.000                       1st Qu.:    0.180
##  Median :43.00   Median : 1.000                       Median :    0.374
##  Mean   :44.05   Mean   : 1.013                       Mean   :  244.619
##  3rd Qu.:52.00   3rd Qu.: 2.000                       3rd Qu.:    0.806
##  Max.   :88.00   Max.   :10.000                       Max.   :21395.000
```

```
##
##  MonthlyIncome    NumberOfOpenCreditLinesAndLoans NumberOfTimes90DaysLate
##  Min.   :     0   Min.   : 0.000                  Min.   : 0.000
##  1st Qu.:  2700   1st Qu.: 3.000                  1st Qu.: 0.000
##  Median :  4182   Median : 6.000                  Median : 0.000
##  Mean   :  5282   Mean   : 6.365                  Mean   : 0.636
##  3rd Qu.:  6430   3rd Qu.: 8.000                  3rd Qu.: 1.000
##  Max.   :141500   Max.   :40.000                  Max.   :15.000
##  NA's   :550
##  NumberRealEstateLoansOrLines NumberOfTime60.89DaysPastDueNotWorse
##  Min.   : 0.0000              Min.   :0.0000
##  1st Qu.: 0.0000              1st Qu.:0.0000
##  Median : 0.0000              Median :0.0000
##  Mean   : 0.6812              Mean   :0.4308
##  3rd Qu.: 1.0000              3rd Qu.:1.0000
##  Max.   :10.0000              Max.   :7.0000
##
##  NumberOfDependents
##  Min.   :0.0000
##  1st Qu.:0.0000
##  Median :0.0000
##  Mean   :0.9204
##  3rd Qu.:2.0000
##  Max.   :8.0000
##  NA's   :61
```

```
summary(g2)
```

```
##        X          SeriousDlqin2yrs  RevolvingUtilizationOfUnsecuredLines
##  Min.   :   294   Min.   :0.00000   Min.   :  112
##  1st Qu.: 43785   1st Qu.:0.00000   1st Qu.: 1082
##  Median : 80200   Median :0.00000   Median : 2159
##  Mean   : 77440   Mean   :0.04933   Mean   : 3848
##  3rd Qu.:110755   3rd Qu.:0.00000   3rd Qu.: 4318
##  Max.   :149280   Max.   :1.00000   Max.   :50708
##
##       age        NumberOfTime30.59DaysPastDueNotWorse   DebtRatio
##  Min.   :24.00   Min.   :0.00                         Min.   :    0.001
##  1st Qu.:39.00   1st Qu.:0.00                         1st Qu.:    0.213
##  Median :48.00   Median :0.00                         Median :    0.381
##  Mean   :50.59   Mean   :0.13                         Mean   :  604.614
##  3rd Qu.:62.50   3rd Qu.:0.00                         3rd Qu.:   81.500
##  Max.   :87.00   Max.   :2.00                         Max.   :21395.000
##
##  MonthlyIncome   NumberOfOpenCreditLinesAndLoans NumberOfTimes90DaysLate
##  Min.   :    0   Min.   : 1.000                  Min.   :0.00000
##  1st Qu.: 4800   1st Qu.: 4.000                  1st Qu.:0.00000
##  Median : 7083   Median : 5.000                  Median :0.00000
##  Mean   : 8629   Mean   : 5.637                  Mean   :0.03139
##  3rd Qu.:10400   3rd Qu.: 7.000                  3rd Qu.:0.00000
##  Max.   :44472   Max.   :21.000                  Max.   :3.00000
##  NA's   :62
##  NumberRealEstateLoansOrLines NumberOfTime60.89DaysPastDueNotWorse
##  Min.   :0.000                Min.   :0.00000
##  1st Qu.:0.000                1st Qu.:0.00000
```

```
##  Median :1.000                Median :0.00000
##  Mean   :1.197                Mean   :0.02242
##  3rd Qu.:2.000                3rd Qu.:0.00000
##  Max.   :9.000                Max.   :1.00000
##
##  NumberOfDependents
##  Min.   :0.000
##  1st Qu.:0.000
##  Median :0.000
##  Mean   :0.684
##  3rd Qu.:1.000
##  Max.   :4.000
##  NA's   :11
```

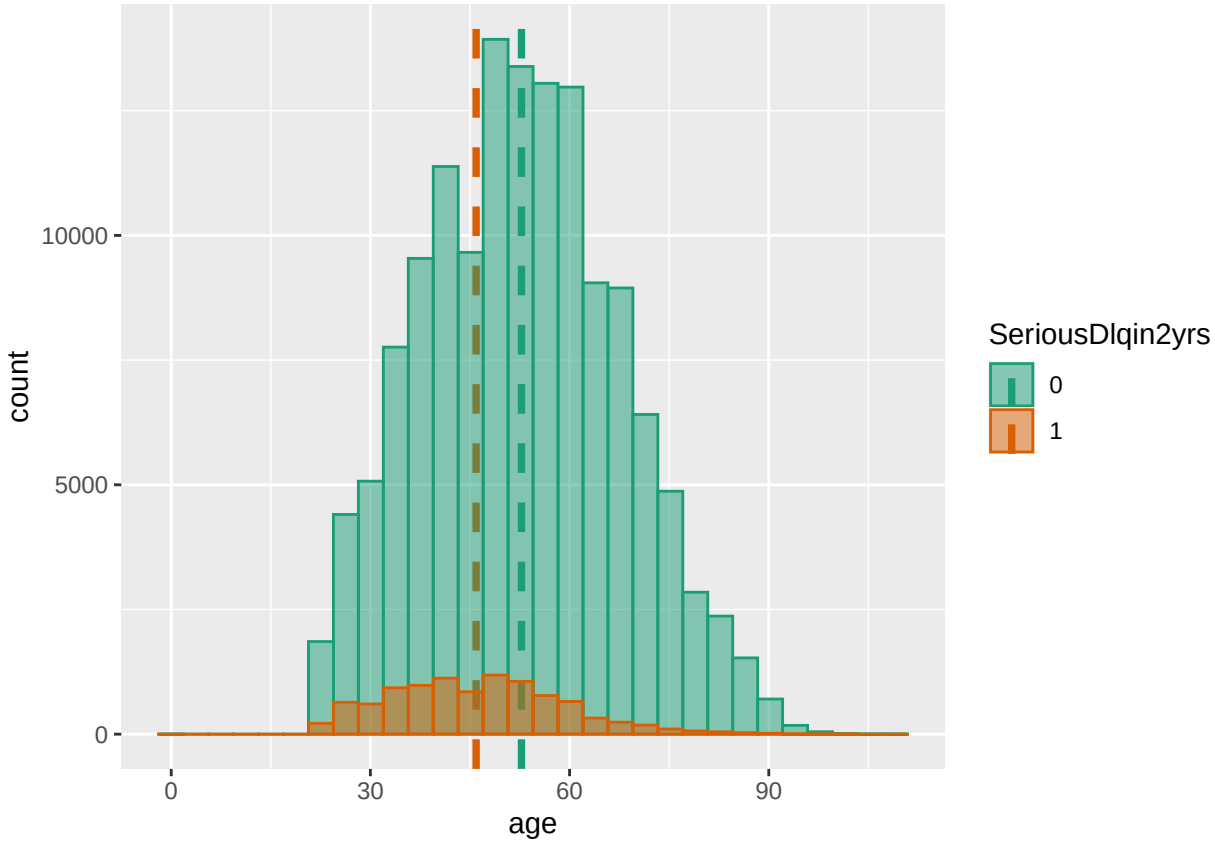
Age

An analysis of the third variable **age** shows that the group who have experienced financial distress in the next two years have an average of age lower than the other group who have not experienced such stress. This may suggest that young people are more likely to experience financial hardships relative to older people. Additionally, the histogram shows that there are far more people who have not experienced any financial distress than the other group.

```
data_train$SeriousDlqin2yrs <- as.factor(data_train$SeriousDlqin2yrs)
mage <- ddply(data_train, "SeriousDlqin2yrs", summarise, grp.mean=mean(age))
head(mage)
```

```
##   SeriousDlqin2yrs grp.mean
## 1                0 52.75138
## 2                1 45.92659
```

```
ggplot(data_train, aes(x=age, color=SeriousDlqin2yrs,
                       fill=SeriousDlqin2yrs)) +
  geom_vline(data=mage, aes(xintercept=grp.mean, color=SeriousDlqin2yrs),
                                  linetype="dashed", size=1.3) +
  geom_histogram(alpha = 0.5, position = "identity") +
  scale_color_brewer(palette="Dark2") +
  scale_fill_brewer(palette="Dark2")
```



NumberOfTime30.59DaysPastDueNotWorse

In this variable, the max number is 98, which is not possible since $98 \times 30 = 2940$ days, which is equivalent to 8 years. However, the variable measures how many times the person has been 30-59 days past dues for the past 2 years, which makes the value 98 impossible. We should remove any value >24.33 as outliers.

Still, looking at the summary statistics stated below, we found that for the group who have experienced financial stress, their mean and standard deviation are both significantly higher than the group who have not.

```
data_train %>%
  group_by(SeriousDlqin2yrs) %>%
  summarise(
          count = n(),
          mean_ntimes = mean(NumberOfTime30.59DaysPastDueNotWorse),
          sd_ntimes = sd(NumberOfTime30.59DaysPastDueNotWorse),
          min_ntimes = min(NumberOfTime30.59DaysPastDueNotWorse),
          max_ntimes = max(NumberOfTime30.59DaysPastDueNotWorse)
            )
```

```
## # A tibble: 2 x 6
##   SeriousDlqin2yrs  count mean_ntimes sd_ntimes min_ntimes max_ntimes
##   <fct>             <int>       <dbl>     <dbl>      <int>      <int>
## 1 0                139974       0.280      2.95          0         98
## 2 1                 10026       2.39      11.7           0         98
```

Data Cleaning

We thought about dropping the missing values, replacing missing values with medians, and using regressions to replace missing values

These code above give us the dataframe with incompleted observations dropped

```
train_complete <- complete.cases(data_train)
data_drop <- cbind(data_train,train_complete)
data_drop <- subset(data_drop,data_drop$train_complete == TRUE)
```

These code gave us data_median which uses the respective medians to fill in NAs in MonthlyIncome and NumberOfDependents