code-550.R

Rishabh Rawat

2022-05-13

```
#Load packages
library(ggplot2)
```

```
## Warning: replacing previous import 'lifecycle::last_warnings' by
## 'rlang::last_warnings' when loading 'tibble'
```

```
## Warning: replacing previous import 'lifecycle::last_warnings' by
## 'rlang::last_warnings' when loading 'pillar'
```

```
library(gridExtra)
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.1.2
```

```
##
## Attaching package: 'dplyr'
```

```
## The following object is masked from 'package:gridExtra':
##
##     combine
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(devtools)
```

```
## Warning: package 'devtools' was built under R version 4.1.3
```

```
## Loading required package: usethis
```

```
## Warning: package 'usethis' was built under R version 4.1.3
```

```
library(GGally)
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

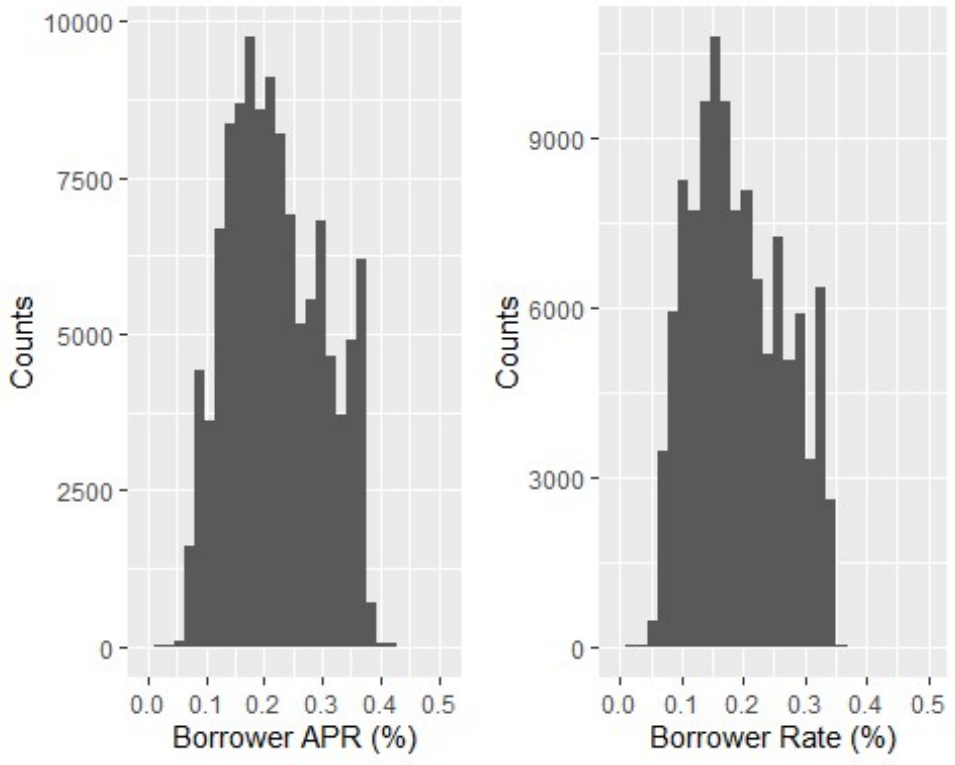
```
#Loading data
loan=read.csv('prosperLoanData.csv')
```

```r
# create new datetime variable for the listing created date
loan$ListingDateTime <- strptime(x = as.character(loan$ListingCreationDate),
                                 format = "%Y-%m-%d")

# The main purposes of this project are to summarize the characteristics of\
# variables that can affect the loan status and to get some ideas about the\
# relationships among multiple variables using summary statistics and data\
# visualizations.


##  BorrowerAPR and BorrowerRate
p1=ggplot(aes(x=BorrowerAPR ), data = loan)+
  geom_histogram()+
  xlab('Borrower APR (%)')+
  ylab('Counts')+
  scale_x_continuous()
p2=ggplot(aes(x=BorrowerRate ), data = loan)+
  geom_histogram()+
  xlab('Borrower Rate (%)')+
  ylab('Counts')+
  scale_x_continuous()
grid.arrange(p1,p2, ncol=2)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 25 rows containing non-finite values (stat_bin).
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

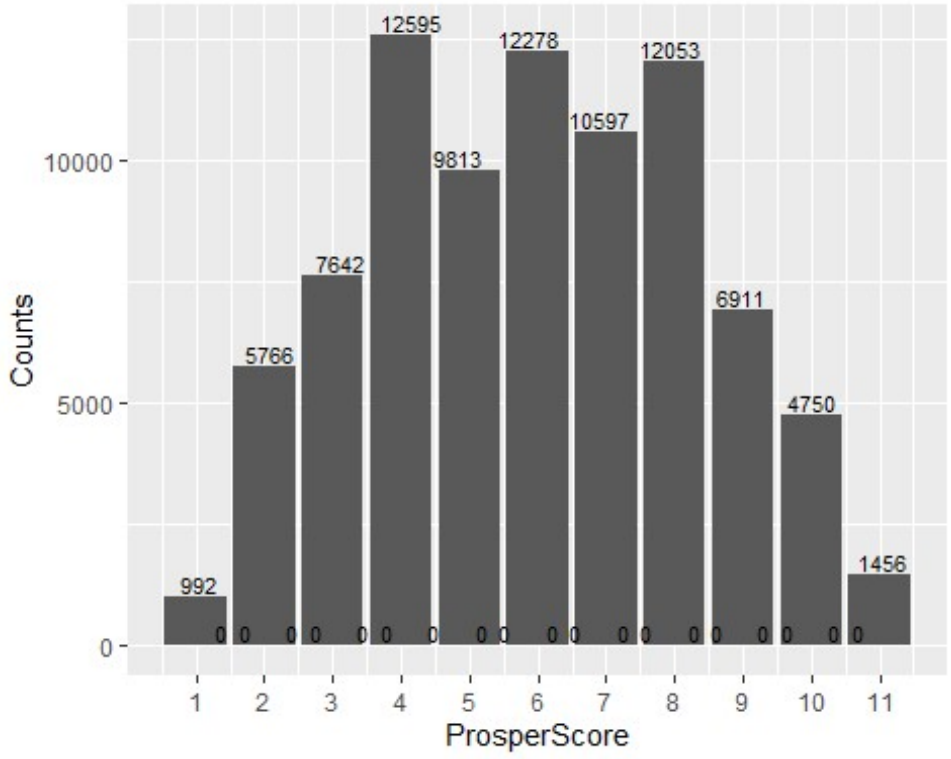



```
ggplot(aes(ProsperScore), data=loan)+
  geom_bar()+
  scale_x_continuous(breaks=c(1:11))+
  geom_text(stat='bin',aes(label=..count..), vjust=-0.2,size=3)+
  ylab('Counts')
```

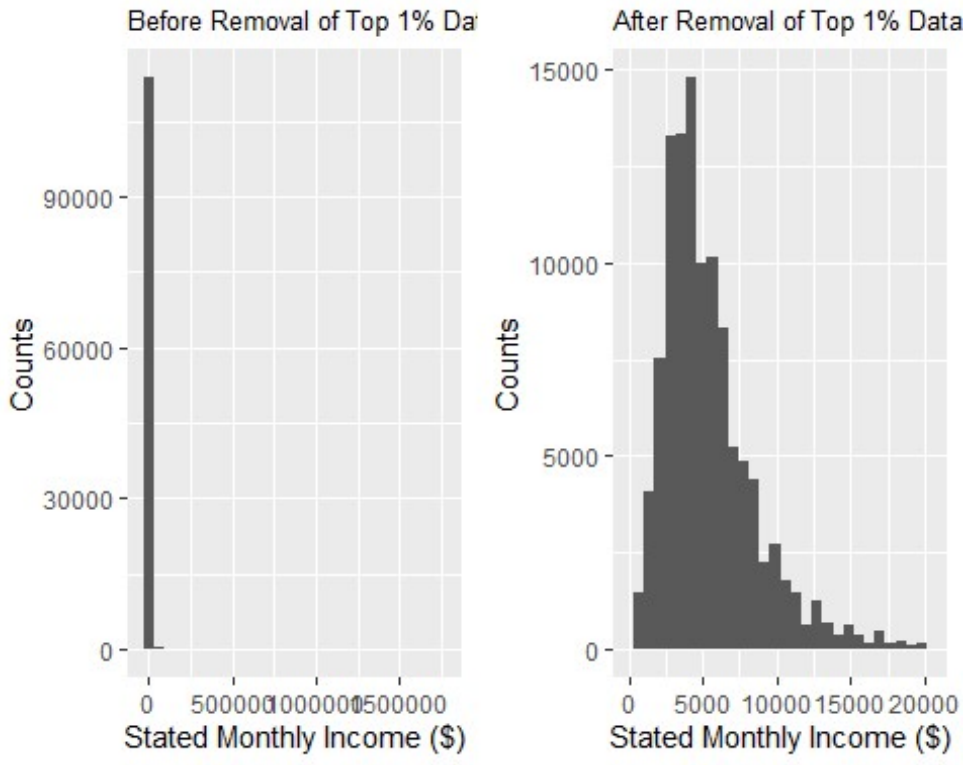
```
## Warning: Removed 29084 rows containing non-finite values (stat_count).
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

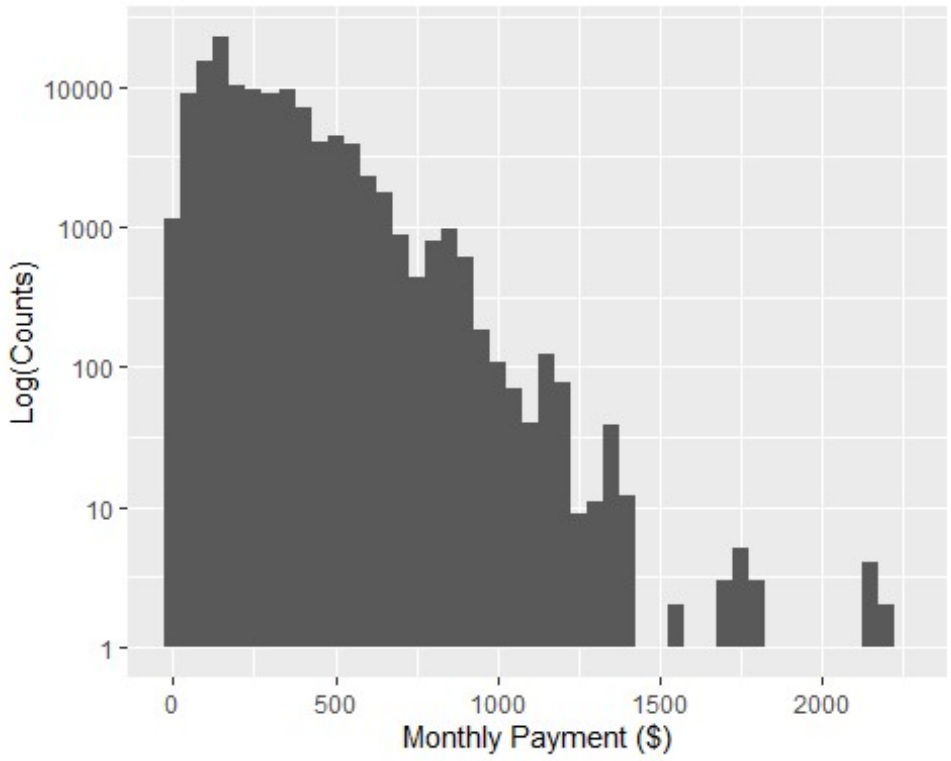
```
## Warning: Removed 29084 rows containing non-finite values (stat_bin).
```



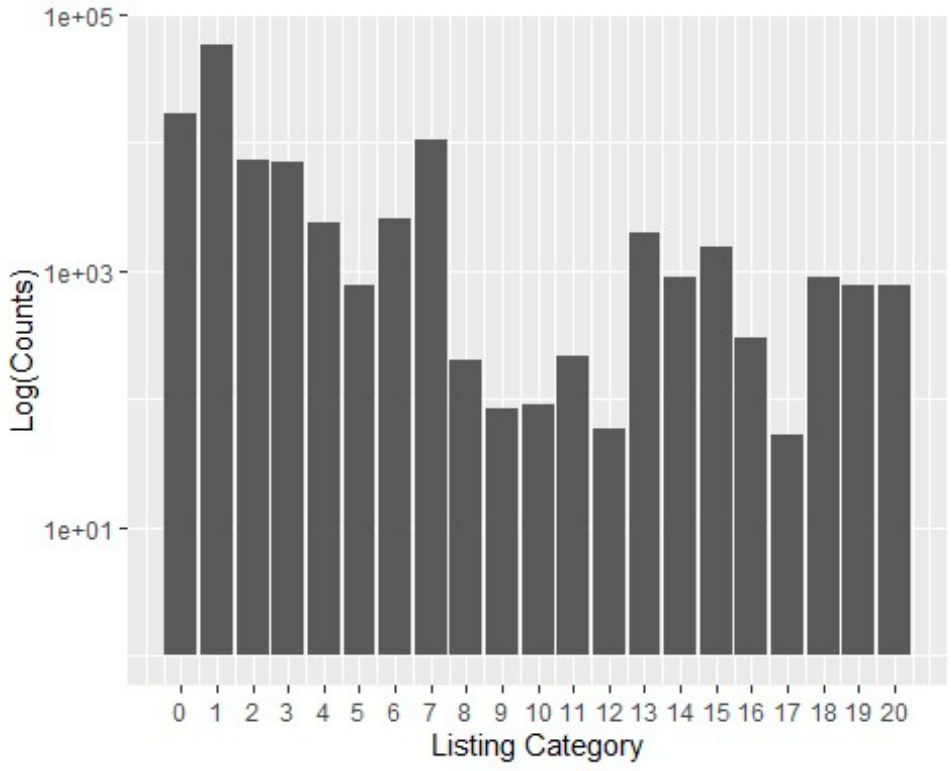

```
## StatedMonthlyIncome

p1=ggplot(aes(x=StatedMonthlyIncome ), data = loan)+
  geom_histogram()+
  scale_x_continuous()+
  ggtitle('Before Removal of Top 1% Data')+
  xlab('Stated Monthly Income ($)')+
  ylab('Counts')+
  theme(plot.title = element_text(size=10))
p2=ggplot(aes(x=StatedMonthlyIncome ), data = loan)+
  geom_histogram()+
  scale_x_continuous(limits = c(0, quantile(loan$StatedMonthlyIncome,0.99)))+
  ggtitle('After Removal of Top 1% Data')+
  xlab('Stated Monthly Income ($)')+
  ylab('Counts')+
  theme(plot.title = element_text(size=10))
grid.arrange(p1,p2, ncol=2)

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 1140 rows containing non-finite values (stat_bin).

## Warning: Removed 2 rows containing missing values (geom_bar).
```




```
## MonthlyLoanPayment

ggplot(aes(x=MonthlyLoanPayment), data=loan)+
  geom_histogram(binwidth = 50)+
  scale_y_log10()+
  xlab('Monthly Payment ($)')+
  ylab('Log(Counts)')

## Warning: Transformation introduced infinite values in continuous y-axis

## Warning: Removed 6 rows containing missing values (geom_bar).
```




```
## ListingCategory
ggplot(aes(x=ListingCategory..numeric.), data=loan)+
  geom_bar()+
  scale_y_log10()+
  scale_x_continuous(breaks=c(0:20))+
  ylab('Log(Counts)')+
  xlab('Listing Category')
```




```
## AvailableBankcardCredit

#Original PLot
p1=ggplot(aes(x=AvailableBankcardCredit), data = loan)+
  geom_histogram()+
  xlab('Bank Card Credits($)')+
  ylab('Counts')+
  scale_x_continuous()+
  theme(axis.text.x=element_text(angle=60, hjust=1, size=6),
        axis.title.x =element_text(size=8))
#LOG transformation
p3=ggplot(aes(x=AvailableBankcardCredit), data = loan)+
  geom_histogram()+
  xlab('Bank Card Credits($)')+
  ylab('Counts')+
  scale_x_log10()+
  theme(axis.text.x=element_text(angle=60, hjust=1, size=6),
        axis.title.x =element_text(size=8))
#Removal of 10% percential
p2=ggplot(aes(x=AvailableBankcardCredit), data =loan)+
  geom_histogram()+
  xlab('Bank Card Credits($)')+
  ylab('Counts')+
  scale_x_continuous(limits=c(0, quantile(loan$AvailableBankcardCredit,
                                          0.99,
                                          na.rm=T)))+
```

```
  theme(axis.text.x=element_text(angle=60, hjust=1, size=6),
        axis.title.x =element_text(size=8) )
grid.arrange(p1,p2,p3,ncol=3)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 7544 rows containing non-finite values (stat_bin).
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

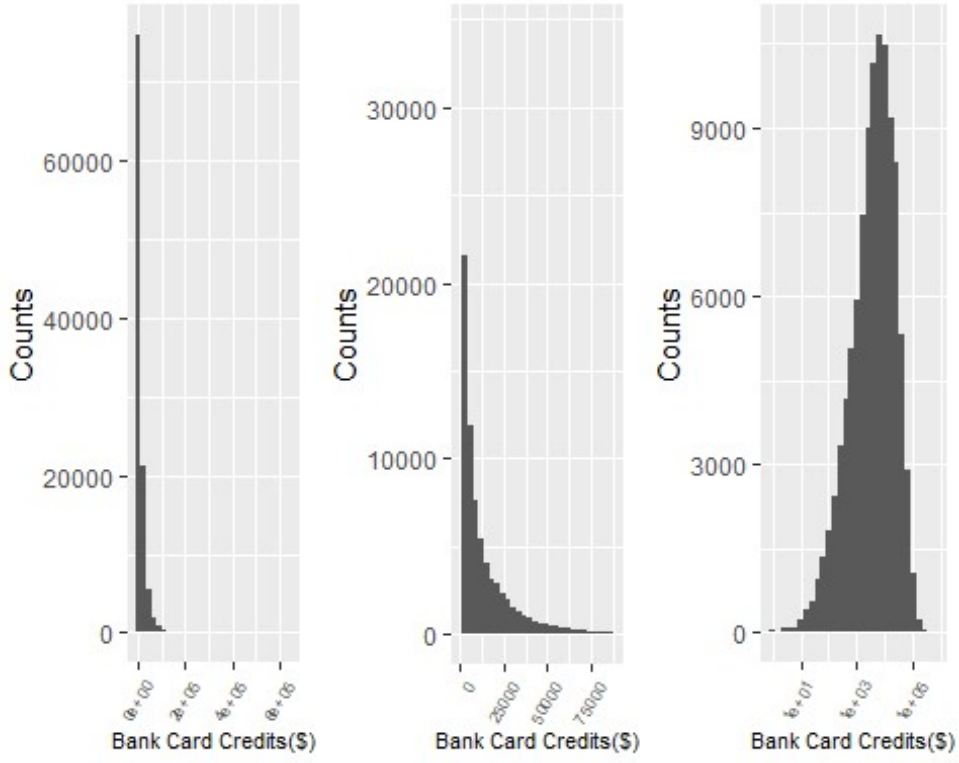
```
## Warning: Removed 8608 rows containing non-finite values (stat_bin).
```

```
## Warning: Removed 2 rows containing missing values (geom_bar).
```

```
## Warning: Transformation introduced infinite values in continuous x-axis
```

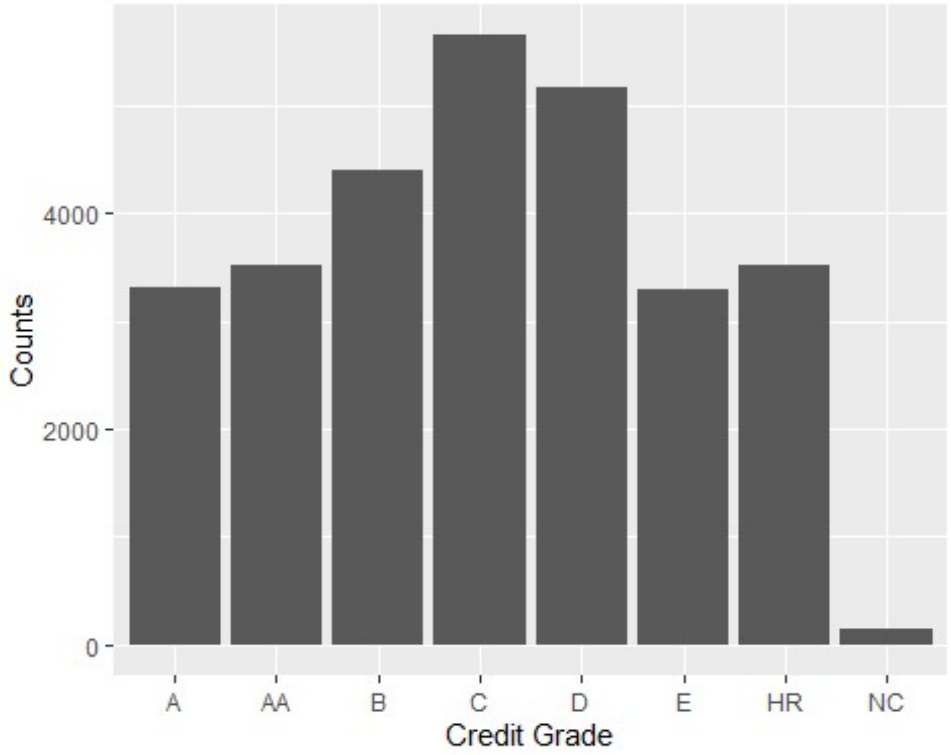
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 12425 rows containing non-finite values (stat_bin).
```




```
## CreditGrade
```

```
CG=subset(loan, loan$CreditGrade !='')
ggplot(aes(x=CreditGrade), data=CG)+
  geom_bar()+
xlab('Credit Grade')+
ylab('Counts')
```




```
ggpairs(loan[,c(9, 4, 20, 29, 34,38,49)])
```

```
## Warning in ggally_statistic(data = data, mapping = mapping, na.rm = na.rm,
:
## Removed 7604 rows containing missing values
```

```
## Warning in ggally_statistic(data = data, mapping = mapping, na.rm = na.rm,
:
## Removed 697 rows containing missing values
```

```
## Warning in ggally_statistic(data = data, mapping = mapping, na.rm = na.rm,
:
## Removed 990 rows containing missing values
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 7604 rows containing non-finite values (stat_boxplot).
```

```
## Warning: Removed 697 rows containing non-finite values (stat_boxplot).
```

```
## Warning: Removed 990 rows containing non-finite values (stat_boxplot).
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

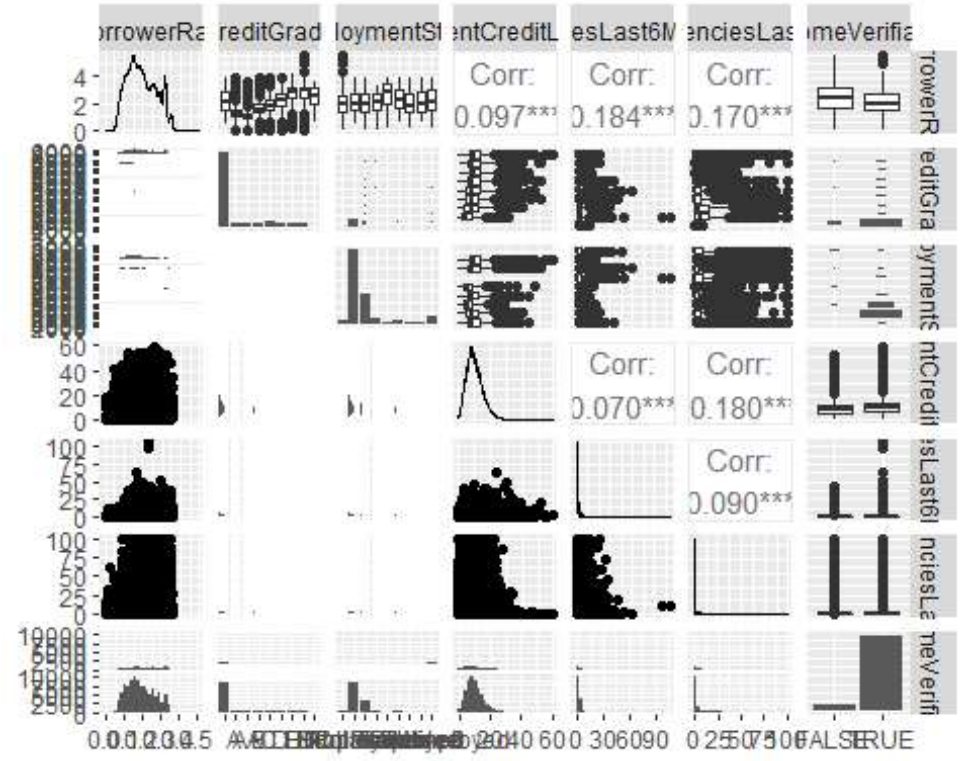
```
## Warning: Removed 7604 rows containing non-finite values (stat_boxplot).
```

```
## Warning: Removed 697 rows containing non-finite values (stat_boxplot).
```

```
## Warning: Removed 990 rows containing non-finite values (stat_boxplot).
```

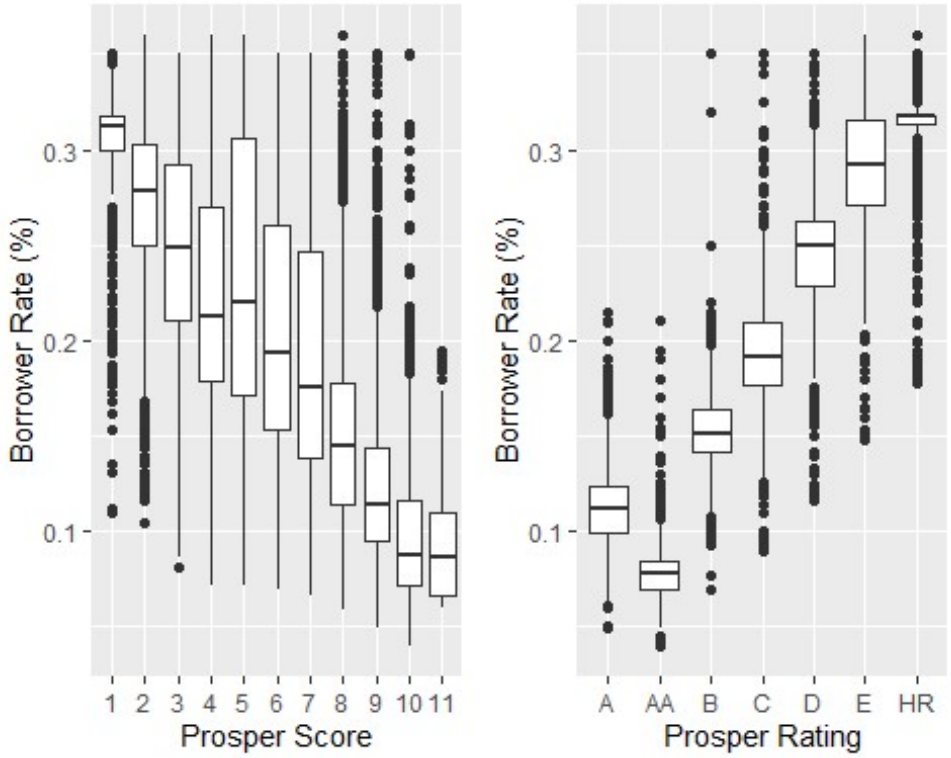
```
## Warning: Removed 7604 rows containing missing values (geom_point).
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## Warning: Removed 7604 rows containing non-finite values (stat_bin).
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## Warning: Removed 7604 rows containing non-finite values (stat_bin).
## Warning: Removed 7604 rows containing non-finite values (stat_density).
## Warning in ggally_statistic(data = data, mapping = mapping, na.rm = na.rm,
:
## Removed 7624 rows containing missing values
## Warning in ggally_statistic(data = data, mapping = mapping, na.rm = na.rm,
:
## Removed 7624 rows containing missing values
## Warning: Removed 7604 rows containing non-finite values (stat_boxplot).
## Warning: Removed 697 rows containing missing values (geom_point).
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## Warning: Removed 697 rows containing non-finite values (stat_bin).
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## Warning: Removed 697 rows containing non-finite values (stat_bin).
## Warning: Removed 7624 rows containing missing values (geom_point).
## Warning: Removed 697 rows containing non-finite values (stat_density).
## Warning in ggally_statistic(data = data, mapping = mapping, na.rm = na.rm,
:
## Removed 990 rows containing missing values
## Warning: Removed 697 rows containing non-finite values (stat_boxplot).
## Warning: Removed 990 rows containing missing values (geom_point).
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## Warning: Removed 990 rows containing non-finite values (stat_bin).
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## Warning: Removed 990 rows containing non-finite values (stat_bin).
## Warning: Removed 7624 rows containing missing values (geom_point).
## Warning: Removed 990 rows containing missing values (geom_point).
```

```
## Warning: Removed 990 rows containing non-finite values (stat_density).

## Warning: Removed 990 rows containing non-finite values (stat_boxplot).

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 7604 rows containing non-finite values (stat_bin).

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 697 rows containing non-finite values (stat_bin).

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 990 rows containing non-finite values (stat_bin).
```




```
## Prosper Rating vs. BorrowerRate

# ProsperScore is the custom risk score built using historical Prosper\
# data. The score ranges from 1-10, with 10 being the best, or lowest risk\
# score. **ProsperRating- is a similar parameter to evaluate a customer's\
# risk to make default loan.
# The plot show strong correlation between the prosper rating/score with the\
# borrower's rate. Generally, borrowers have no clue on their prosper
scores.\
# So these information can not be included in our model.
```

```
p1=ggplot(aes(x=ProsperRating..Alpha., y=BorrowerRate),
          data=subset(loan, loan$ProsperRating..Alpha.!=''))+
  geom_boxplot()+
  xlab('Prosper Rating')+
  ylab('Borrower Rate (%)')
p2=ggplot(aes(x=as.factor(ProsperScore), y=BorrowerRate),
          data=subset(loan, loan$ProsperRating..Alpha.!=''))+
  geom_boxplot()+
  xlab('Prosper Score')+
  ylab('Borrower Rate (%)')
grid.arrange(p2,p1, ncol=2)
```




```
## AvailableBankcardCredit vs. BorrowerRate

# AvailableBankcardCredit is total available credit via bank card.\
# It can be an indicator of a borrower's credit history.
# A reasonable guess is that a higher available bank card credit will lower\
# your interest rate. This guess is proved to be true from following plot.\
# The points are clustered at the bottom left corner, when a borrower has\
# relatively low credit amount (<25,000) the probabilities of getting low\
# and high interest rates are similar. However, when the borrower's credit\
# amount is high (>50,000), he/she is more likely to get a lower interest
rate.

ggplot(aes(x=AvailableBankcardCredit, y=BorrowerRate),
```

```
      data=subset(loan,
                  loan$AvailableBankcardCredit<
                    quantile(loan$AvailableBankcardCredit, 0.99, na.rm=T)))+
  geom_point(alpha=0.05, color="#cf4c35")+
   geom_smooth(method=lm, colour="black", size=0.2)+
  xlab('Available Bankcard Credit ($)')+
  ylab('Borrower Rate (%)')
```

```
## `geom_smooth()` using formula 'y ~ x'
```