Initial Loan Book Analysis

Thomas Hughes

The following is an analysis of Lending Club Loan Data.

Libraries

Let's start by importing the necessary libraries:

```
library(ggplot2)
library(tidyverse)
library(data.table)
library(DescTools)
library(caTools)
```

Import

Next we'll import the file. Note that this is a rather large file. It has over 2 million rows and about 150 variables. To work efficiently, we'll use data.table.

Many of the variables are empty or of little value to our analysis, so we will import only those which are useful:

```
loans <- fread(input = "loan.csv", select = c("loan_amnt",
                                              "term",
                                              "int_rate",
                                              "grade",
                                              "emp_title",
                                              "emp_length",
                                              "home_ownership",
                                              "annual_inc",
                                              "issue_d",
                                              "loan_status",
                                              "purpose",
                                              "title",
                                              "dti",
                                              "delinq_2yrs"))
```

Cleaning

First let's get rid of any NA values:

```
loans <- na.omit(loans, invert = F)
```

We assume that any dti > 100 is reported in error, and thus remove them:

```
loans <- loans[dti < 100 | dti == 100,]
```

We convert the issue_d column from a chr object to a Date object. It is missing day information so that we insert a place-holder ourselves:

```
loans <- loans[, issue_d := as.Date(stringr::str_replace(loans[, issue_d], "^", "01-"), format = "%d-%b
```

We'll factor grade:

```
loans <- loans[, grade := as.factor(grade)]
```

We create a column which indicates whether a loan is good or bad depending on its status. First we create a vector, badLoans, of bad loan types:

```
badLoans <- c("Charged Off", "Default", "Does not meet the credit policy. Status:Charged Off",
              "In Grace Period", "Late (16-30 days)", "Late (31-120 days)")
```

Next we write a function which classifies the loan using badLoans:

```
classifyLoan <- function(s){
  if(s %in% badLoans){
    return(0)
  }else{
    return(1)
  }
}
```

Now, we add a new column, loan_status_simple, to loans indicating the loan type:

```
loans <- loans[, loan_status_simple := sapply(loans[,loan_status], classifyLoan, USE.NAMES = F)]
```

Finally, we factor our simple status:

```
loans <- loans[, loan_status_simple := factor(loans[,loan_status_simple], labels = c("Bad", "Good"))]
```

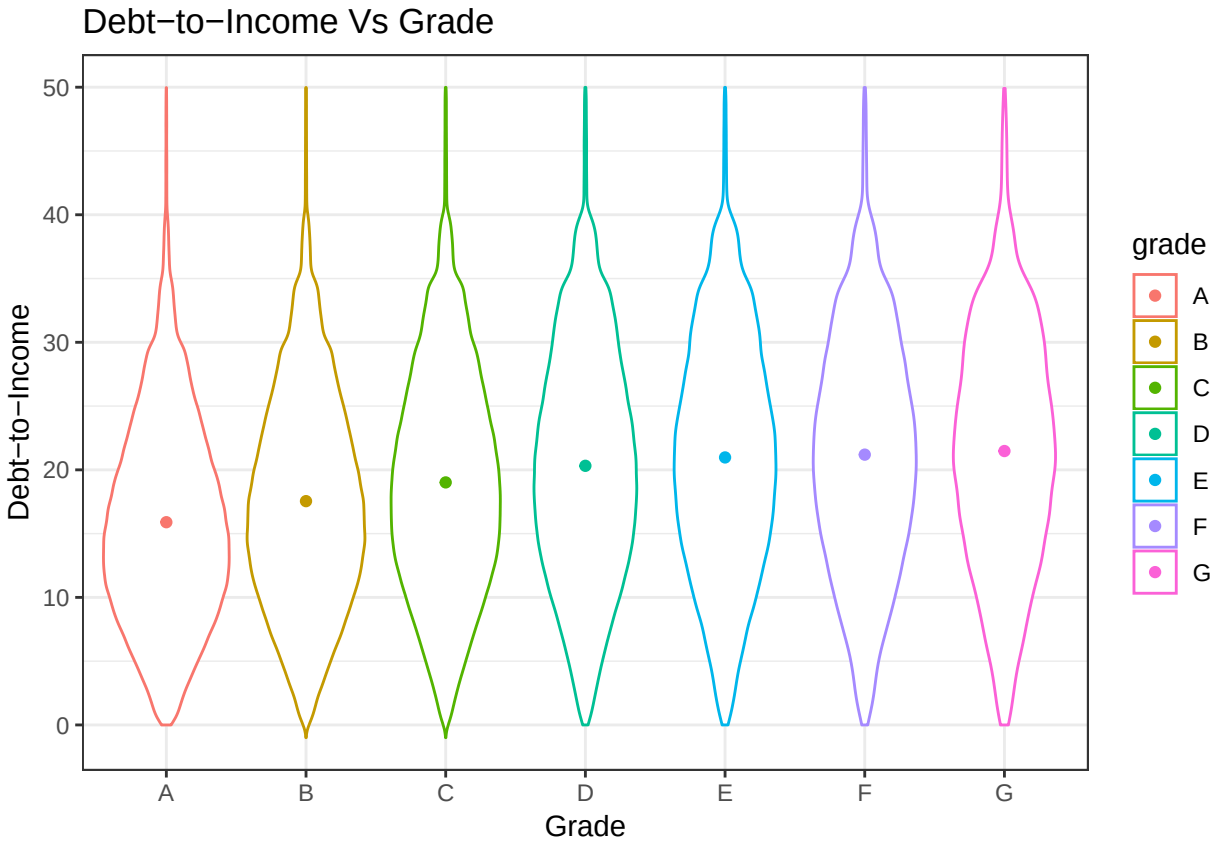
Exploratory Analysis

Debt-to-Income

Let's create a violin plot of Debt-to-Income Vs. Grade: *(99% of all values fall below 50 so we restrict our graph to Debt-to-Income ratios below 50)*

```
dtiVsGradePlot <- ggplot(loans[dti < 50,dti, by = grade],aes(x = grade, y = dti, color = grade)) +
  geom_violin() +
  stat_summary(fun.y = "mean", geom = "point") +
  labs(x = "Grade", y = "Debt-to-Income",
       title = "Debt-to-Income Vs Grade") +
  theme_bw()

plot(dtiVsGradePlot)
```



We can see that higher the debt-to-income ratio, the riskier loan, which suggests healthy lending practices.

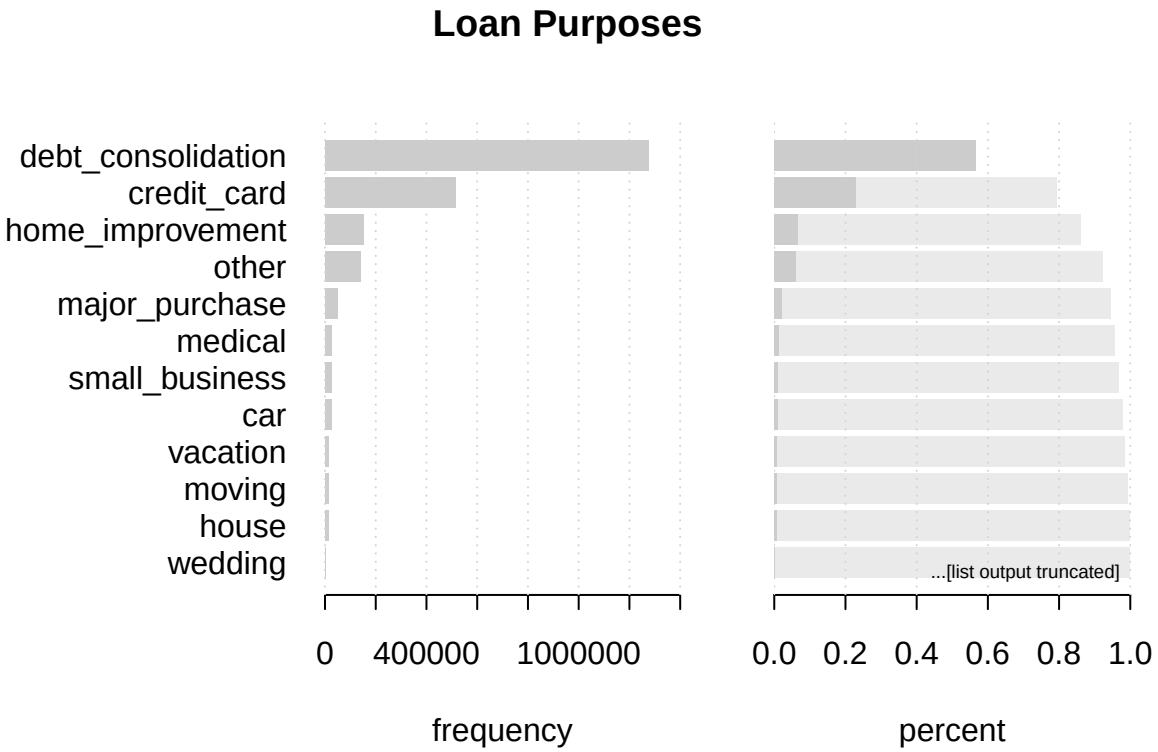
Loan Purposes

Let's plot loan purposes:

```
Desc(loans[,purpose], main = "Loan Purposes", plotit = T)
```

```
## ------------------------------------------------------------------------
## Loan Purposes
##
##   length      n    NAs unique levels  dupes
##    2e+06  2e+06      0  1e+01  1e+01      y
##          100.0%   0.0%
##
##                   level   freq   perc  cumfreq  cumperc
## 1   debt_consolidation  1e+06  56.5%    1e+06    56.5%
## 2          credit_card  5e+05  22.9%    2e+06    79.4%
## 3     home_improvement  2e+05   6.7%    2e+06    86.0%
## 4                other  1e+05   6.2%    2e+06    92.2%
## 5       major_purchase  5e+04   2.2%    2e+06    94.4%
## 6              medical  3e+04   1.2%    2e+06    95.7%
## 7       small_business  2e+04   1.1%    2e+06    96.8%
## 8                  car  2e+04   1.1%    2e+06    97.8%
## 9              vacation  2e+04   0.7%    2e+06    98.5%
## 10               moving  2e+04   0.7%    2e+06    99.2%
## 11                house  1e+04   0.6%    2e+06    99.8%
```

```
## 12             wedding  2e+03   0.1%    2e+06    99.9%
## ... etc.
##  [list output truncated]
```



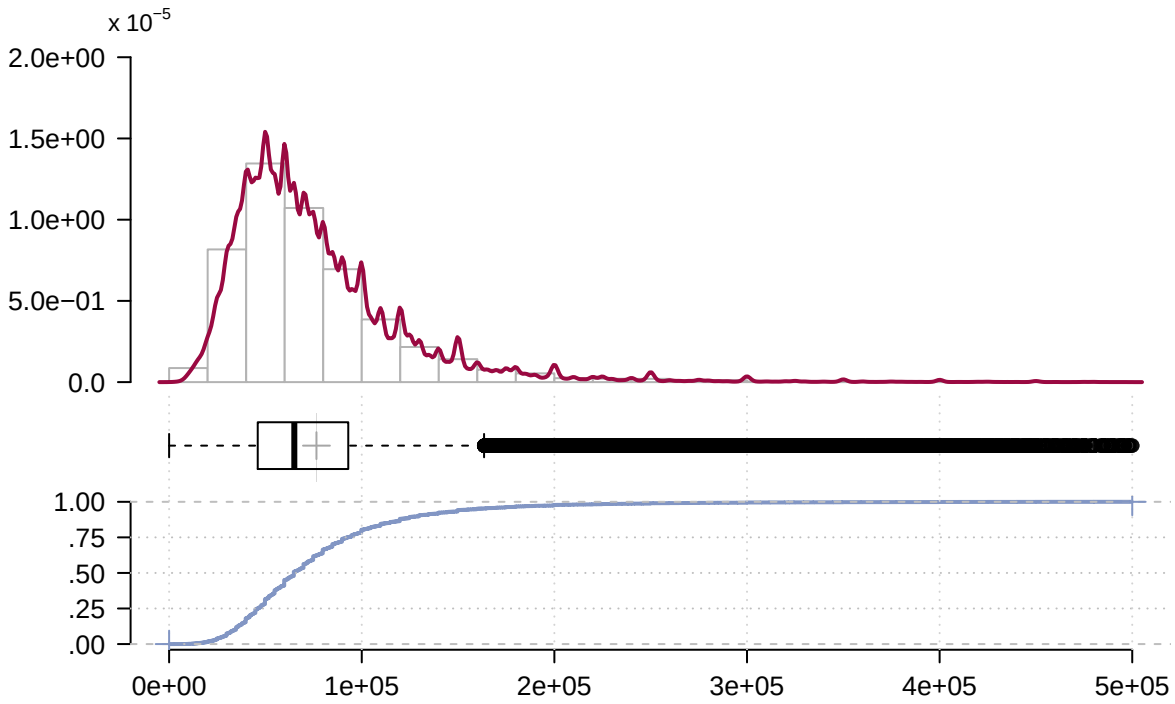
Debt consolidation was the most popular reason for taking out a loan.

Annual Income

Let's create a density plot of distribution of annual income: *(99% of incomes fall below $500,000.00 so we restrict our graph to these values)*

```
Desc(loans[annual_inc < 500000,annual_inc], main = "Annual Income", plotit = T)
```

```
## ------------------------------------------------------------------------
## Annual Income
##
##     length         n       NAs    unique        0s      mean    meanCI
##      2e+06     2e+06         0     9e+04     8e+00  7.65e+04  7.64e+04
##             100.0%      0.0%               0.0%            7.66e+04
##
##        .05       .10       .25    median       .75       .90       .95
##   2.80e+04  3.40e+04  4.60e+04  6.50e+04  9.30e+04  1.30e+05  1.60e+05
##
##      range        sd     vcoef       mad       IQR      skew      kurt
##   5.00e+05  4.69e+04  6.13e-01  3.26e+04  4.70e+04  2.45e+00  1.04e+01
##
## lowest : 0.0 (8e+00), 4.30e+02, 6.00e+02 (2e+00), 6.40e+02, 6.86e+02
## highest: 4.97e+05 (2e+00), 4.98e+05 (9e+00), 4.99e+05, 4.99e+05 (4e+00), 5.00e+05 (4e+00)
```

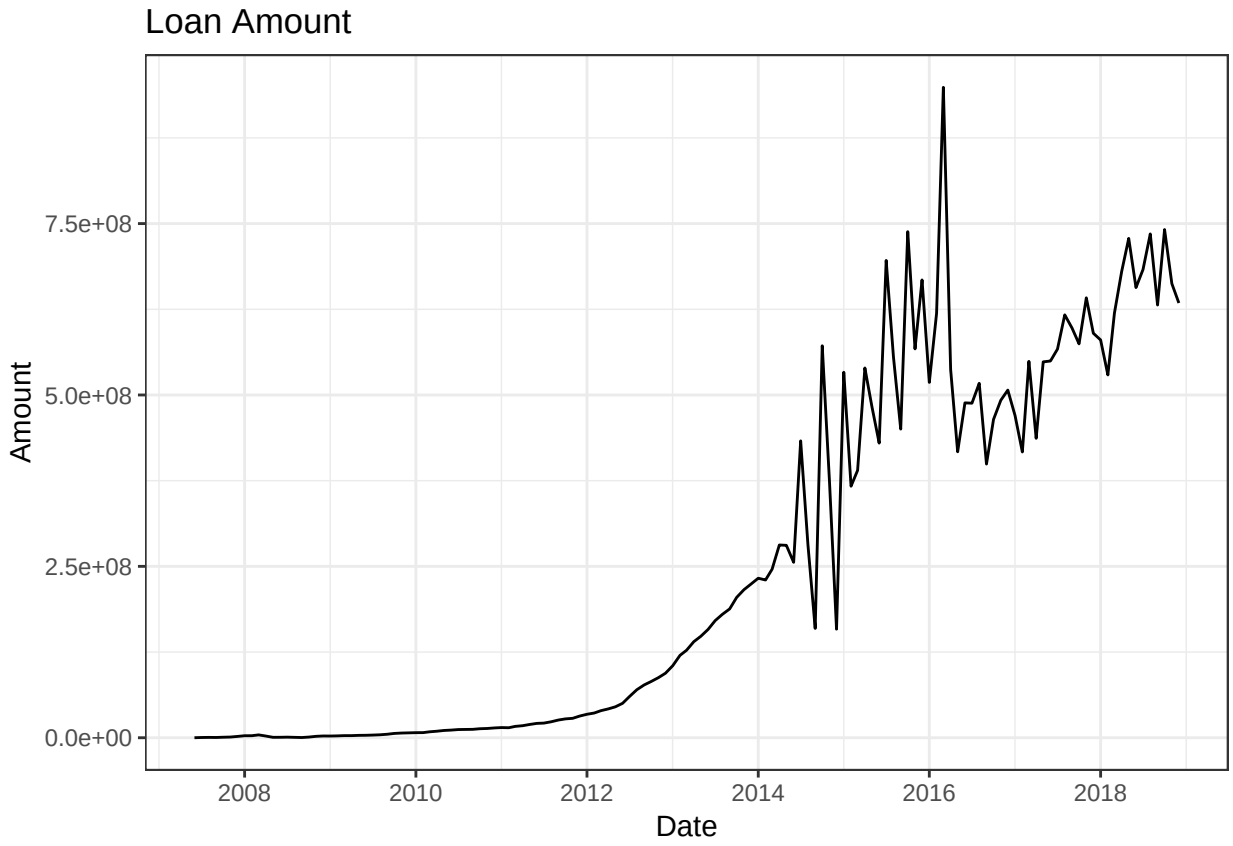


This suggests our population of borrowers is not a representative sample of income earners nationwide, as their median wage is slightly above the national median over the same time period.

Loan Amounts Time Series

First let's look at the total amounts lent as a time series:

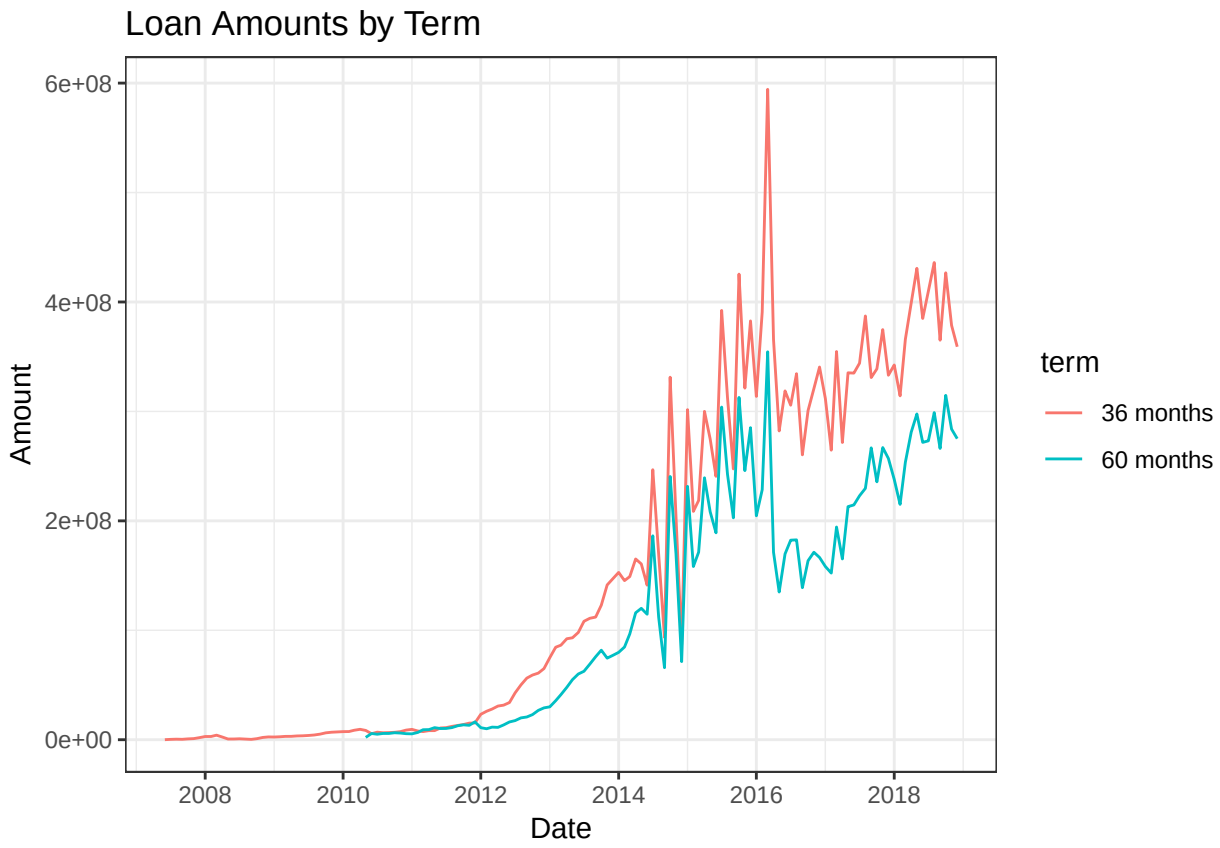
```
loanAmntTsPlot <- ggplot(loans[,.(Amount = sum(loan_amnt)), by = issue_d], aes(x = issue_d, y = Amount))
  geom_line() +
  theme_bw() +
  labs(x = "Date",
       title = "Loan Amount")
print(loanAmntTsPlot)
```



There seems to be a clear bifurcation point around the middle of 2014 which indicates a change in lending practices. There is clear growth.

Next let's look at loan amounts by term:

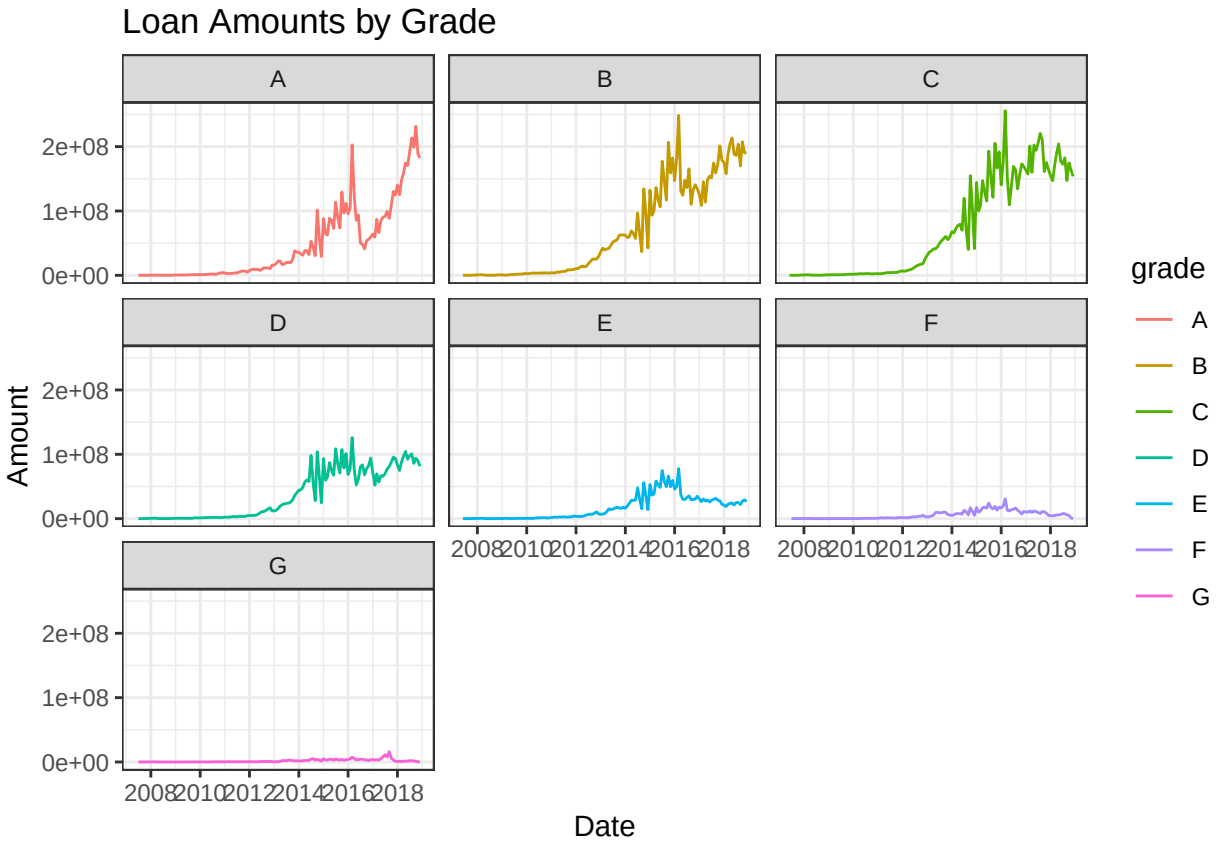
```
loanAmntTermTsPlot <- ggplot(loans[,.(Amount = sum(loan_amnt)), by = .(issue_d, term)], aes(x = issue_d
  geom_line() +
  theme_bw() +
  labs(x = "Date",
       title = "Loan Amounts by Term")
print(loanAmntTermTsPlot)
```



Most loans are made on a 36 month term. It is interesting to note that in most cases, it seems the 60 month term loans closely track.

Next we'll look at loan amounts by grade:

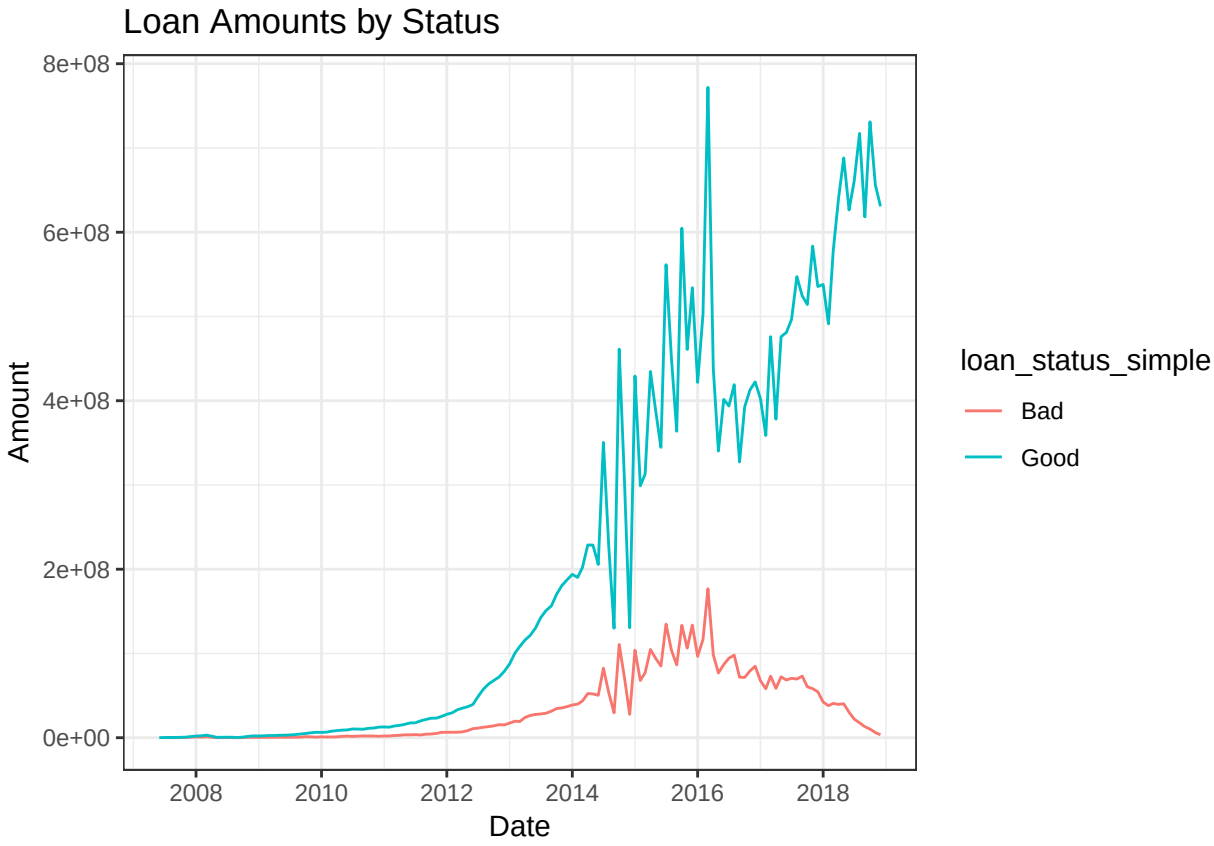
```
loanAmntGradeTsPlot <- ggplot(loans[,.(Amount = sum(loan_amnt)), by = .(issue_d, grade)], aes(x = issue_
  geom_line() +
  facet_wrap(vars(grade)) +
  theme_bw() +
  labs(x = "Date",
       title = "Loan Amounts by Grade")
print(loanAmntGradeTsPlot)
```



The bulk of loan amounts comes from low risk lows. The beginning of 2016 saw a drop in grade A loans, but has since picked back up. It might be interesting to see what happened.

Finally, we'll look at loan amounts by status:

```
loanAmntStatusTsPlot <- ggplot(loans[,.(Amount = sum(loan_amnt)), by = .(issue_d, loan_status_simple)],
  geom_line() +
  theme_bw() +
  labs(x = "Date",
       title = "Loan Amounts by Status")
print(loanAmntStatusTsPlot)
```



The majority of loans are performing well. Even better, lending volume doesn't necessarily seem to imply higher volumes of low-performing loans. After 2018 lending volume increases while amounts on low-performing loans decrease.

Machine Learning

We would like to predict when a loan will be good or bad. To that end, we will build a logistic model to try to predict when a loan will perform well or poorly.

First we'll split our data.table into training and testing data groups:

```
split <- caTools::sample.split(loans$loan_status_simple, SplitRatio = 0.50)
logisticTraining <- subset(loans, split == T)
logisticTesting <- subset(loans, split == F)
```

Next we'll create our logistic model:

```
logisticModel <- glm(loan_status_simple ~ loan_amnt
                     + int_rate + grade + annual_inc
                     + dti + delinq_2yrs,
                     family = binomial(link = "logit"),
                     data = logisticTraining)
```

Now we'll test our model against testing data:

```
fittedProbabilites <- predict(logisticModel,
                              logisticTesting,
                              type = 'response')
fittedResults <- ifelse(fittedProbabilites>0.5, "Good", "Bad")
```

Let's see how well model is performing:

```
error <- mean(fittedResults != logisticTesting$loan_status_simple)
print(1-error)
```

```
## [1] 0.8684481
```

```
table(logisticTesting$loan_status_simple, fittedResults)
```

```
##       fittedResults
##           Bad   Good
##   Bad      47 148312
##   Good    102 979717
```

Conclusion

Our model is able predict testing data with roughly 87% accuracy on a consistent basis. It is worth noting that, unsurprisingly, the model does fairly well predicting when loans will perform well, but is bad at determing when loans will perform poorly. This is likely due to biased data.

We may try a partial-augmentation to the bad loan data to try balancing out the data.