The background is a dark navy blue. It features several large, overlapping, semi-transparent geometric shapes in various colors: bright green, cyan, magenta, orange, and red. These shapes are arranged in a way that creates a sense of depth and movement, with some appearing to recede and others to come forward.

MAKING SENSE OF \$1.3T WORTH OF STUDENT LOAN DEBT



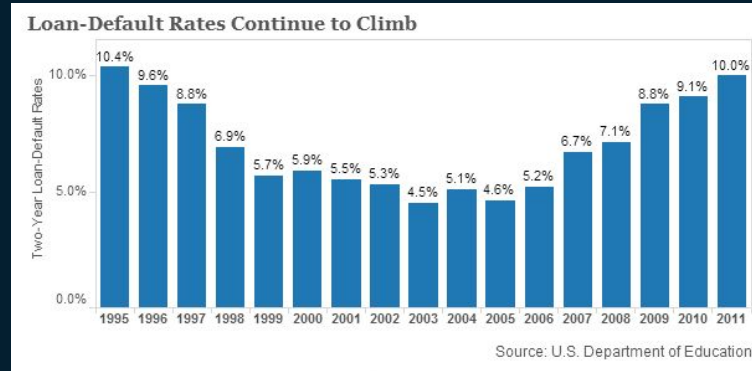
Trevor Ford

Senior Data Analyst @ Student Loan Hero

<https://github.com/trevcodes>

The Challenge & Objective

After a period of decline, default rates amongst student loan borrowers have been on the rise since 2005.



Defaulting on student loans has a number of consequences on a borrower's financial health including loss of eligibility for loan forgiveness programs, lowered credit score (with its own set of consequences), wage garnishment, collections fees, legal action, and should the borrower ever wish to refinance those loans, likely higher interest rates.

If we can predict who is most likely to default on their student loans before they default on their student loans, Student Loan Hero might be able to help borrowers avert default and find alternatives to aid borrowers in their unique situation.

Data Sets

In order to best understand the likelihood of any given borrower is to default on their student loan obligations, two unique data sets were used.



Student Loan Hero

Student Loan Hero App Users (Private)

An anonymized data set of all users who signed up for a proprietary app designed to analyze a user's student loans and provide a customized repayment plan.

This data set contains 20,000+ unique records with 30+ factors.



Carnegie University Classifications (Public)

A data set containing empirical data for all U.S. higher education institutions including facets such as size and settings of institutions, enrollment profile, degree types offered, for-profit or not-for-profit status, along with over 90 other distinguishable variables.

Carnegie University Classifications Can be accessed at the following address:
<http://carnegieclassifications.iu.edu/downloads.php>

Cleaning the SLH App User Data Set

The SLH app user data set was an untidy data set on first load. There were numerous duplicative or unstandardized naming conventions within a number of factors. Date factors used several formats, loan status had nearly 150 different levels (there are less than 10 different loan statuses).

```
> str(slhloans)
'data.frame':   36073 obs. of  24 variables:
 $ User.ID..      : int  41009 41009 41009 41009 41009 41009 37345 37345 44785 44785 ...
 $ User.DOB       : Factor w/ 1361 levels "", "1/1/1954",...: 1142 1142 1142 1142 1142 1142 267 267 1 1 ...
 $ Loan.ID..      : int  377262 377263 377261 377260 377259 377266 346115 346114 421088 421090 ...
 $ Original.Principal : num  2000 3500 4000 4500 6000 10000 2400 9000 2000 2000 ...
 $ Current.Principal : num  2000 3500 4000 4500 6000 ...
 $ Rate           : num  3.86 3.86 4.66 4.66 4.66 6.9 6 5 6.55 6.55 ...
 $ Loan.Disbursement.Date : Factor w/ 3490 levels "", "1/1/2001",...: 1210 1210 2795 514 514 1 1891 1230 3450 3472 ...
 $ Monthly.Interest : num  6.43 11.26 15.53 17.48 23.3 ...
 $ Monthly.Payment  : num  20.1 35.2 41.8 47 62.6 ...
 $ Name             : Factor w/ 3072 levels " \tCitiAssistÂ® Undergraduate Loan",...: 1876 1875 1876 1875 1876 2857 2924 2924 1876 1876 ...
 $ Status           : Factor w/ 149 levels "ADMIN DELINQ PRIOR TO IDR",...: 101 101 101 101 101 76 104 104 87 87 ...
 $ Type             : Factor w/ 2 levels "Fixed","Variable": 1 1 1 1 1 1 2 2 1 1 ...
 $ Education.Degree : Factor w/ 6 levels "Associates","Bachelors",...: 1 1 1 1 1 1 2 2 2 2 ...
 $ College          : Factor w/ 1686 levels " Central Washington University",...: 1001 1001 1001 1001 1001 1001 303 303 559 559 ...
 $ Major            : Factor w/ 1340 levels " health science",...: 777 777 777 777 777 777 1130 1130 483 483 ...
 $ Employment.Status : Factor w/ 6 levels "Employed full-time",...: 1 1 1 1 1 1 6 6 1 1 ...
 $ Adjusted.Gross.Income : num  7 7 7 7 7 25 25 27.5 27.5 ...
 $ Joint.Federal.Income.Tax. : logi  NA NA NA NA NA NA ...
 $ Spouse.Adjusted.Gross.Income: num  NA NA NA NA NA NA NA NA 24000 24000 ...
 $ Employer.Type      : Factor w/ 5 levels "Employee of a for-profit company",...: 3 3 3 3 3 3 4 4 2 2 ...
 $ Profession         : Factor w/ 1854 levels "", " ", " ",...: 414 414 414 414 414 414 1 1 349 349 ...
 $ Family.Size        : Factor w/ 7 levels "1","2","3","4",...: 4 4 4 4 4 4 3 3 2 2 ...
 $ State.of.Residency : Factor w/ 53 levels "AK","AL","AR",...: 49 49 49 49 49 49 10 10 3 3 ...
 $ Credit.Score       : Factor w/ 6 levels "Average (620-679)",...: 5 5 5 5 5 5 1 1 3 3 ...
```


Cleaning the SLH App User Data Set

Loan Status

The SLH app user data set had 149 different loan statuses, many of which were duplicative, misspellings, or included user-specific dates.

Since our end goal of this project is to use a binary classification algorithm to determine the likelihood of default, we needed exactly two different statuses for any given loan: Default or Repayment.

```
Factor w/ 149 levels "ADMIN DELINQ PRIOR TO IDR",...: 101 101 101 101 101 76 104 104 87 87 ...  
> head(levels(factor(slhloans$status)))  
[1] "ADMIN DELINQ PRIOR TO IDR"          "ADMIN FORBEARANCE"  
[3] "ADMIN PRE-HARDSHIP FORB"          "ADMINISTRATIVE FORBEARANCE"  
[5] "ADMINISTRATIVE FORBEARANCE-ENDS 03/08/2016" "ADMINISTRATIVE FORBEARANCE-ENDS 11/15/2016"
```



```
> levels(factor(slhloans$status))  
[1] "Default" "RPM"
```

Cleaning the SLH App User Data Set

Profession

The SLH app user data set had over 1,800 different professions, the majority of which were duplicative or a subset of a larger profession.

Since our end goal of this project is to use a binary classification algorithm to determine the likelihood of default, we needed a significantly reduced number of professions. After deduping and bucketing appropriate professions, we settled on reducing the factor to the 10 most frequently occurring professions and an "Other" bucket for any profession that did not fit within those 10.

```
[961] "M&E analyst"
[963] "Machine Operator"
[965] "Maintenance Supervisor"
[967] "Management"
[969] "Management Consultant (Analyst)"
[971] "Management Engineer"
[973] "Management Trainee"
[975] "Manager"
[977] "Manager of Training and Technical Assistance"
[979] "Manager of US Operations"
[981] "manufacturer "
[983] "Manufacturing "
[985] "Manufacturing Manager"
[987] "Marine and Aquatic Biologist"
[989] "Marine Mammal Scientist"
[991] "Marketer"
[993] "Marketing "
[995] "Marketing and Advertising Account Executive"
[997] "Marketing Communications Specialist "
[999] "Marketing Coordinator"
[ reached getOption("max.print") -- omitted 854 entries ]
"M.D. "
"Machinest"
"Managed Care Clinical Support Coordinator"
"Management Assistant"
"Management Consulting"
"Management in trainee"
"manager"
"Manager "
"Manager of Transition Services"
"Manager, Digital Scheduling"
"Manufacturing"
"Manufacturing Engineer"
"Manufacturing Ops Associate"
"Marine Electrician"
"Market Research Analyst"
"Marketing"
"Marketing and Advertising"
"Marketing Associate"
"Marketing Consultant"
"Marketing Coordinator "
```

[1] "General Business"	"Teacher"	"Doctor/Nurse/Pharmacist"	"Accountant"	"Attorney"
[6] "Engineer"	"Computer/Tech"	"Designer"	"Student"	"Retail"
[11] "other"				

Cleaning the SLH App User Data Set

Major

The SLH app user data set had over 1,300 different majors, the majority of which were duplicative.

Since our end goal of this project is to use a binary classification algorithm to determine the likelihood of default, we needed a significantly reduced number of majors. After deduping and bucketing appropriate majors, we settled on reducing the factor to the 11 most frequently occurring majors and an "Other" bucket for any major that did not fit within those 11.

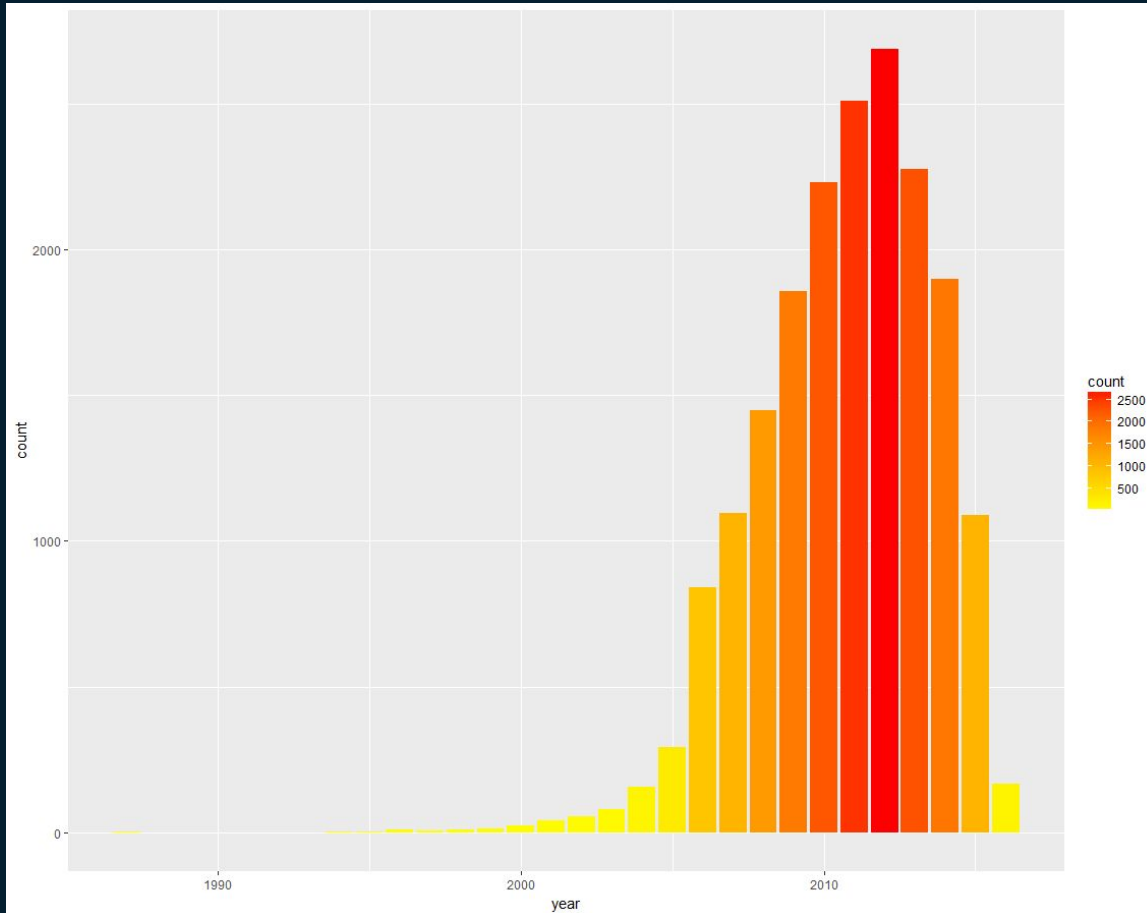
```
[982] "Music History & English"  
[983] "Music Industry"  
[984] "Music Performance"  
[985] "Music Technology"  
[986] "Music Technology "  
[987] "Music Theory"  
[988] "Musical Theatre"  
[989] "N/A"  
[990] "Natural Resources Planning and Decision Making"  
[991] "Naturopathic Medicine, Acupuncture"  
[992] "Near East Studies"  
[993] "Network & Communications Management"  
[994] "Network Administration"  
[995] "Network communications and management"  
[996] "Network Communications Management"  
[997] "Network Computer Systems"  
[998] "Network Engineering and Administration"  
[999] "Network Security"  
[1000] "Network Systems administration"  
[ reached getOption("max.print") -- omitted 340 entries ]
```



[1] "Law"	"Business"	"Engineering"	"Sciences"	"Education"	"Psychology"
[7] "Liberal Artss"	"Communications"	"Higher Medical Degree"	"Nursing"	"MBA"	"Other"

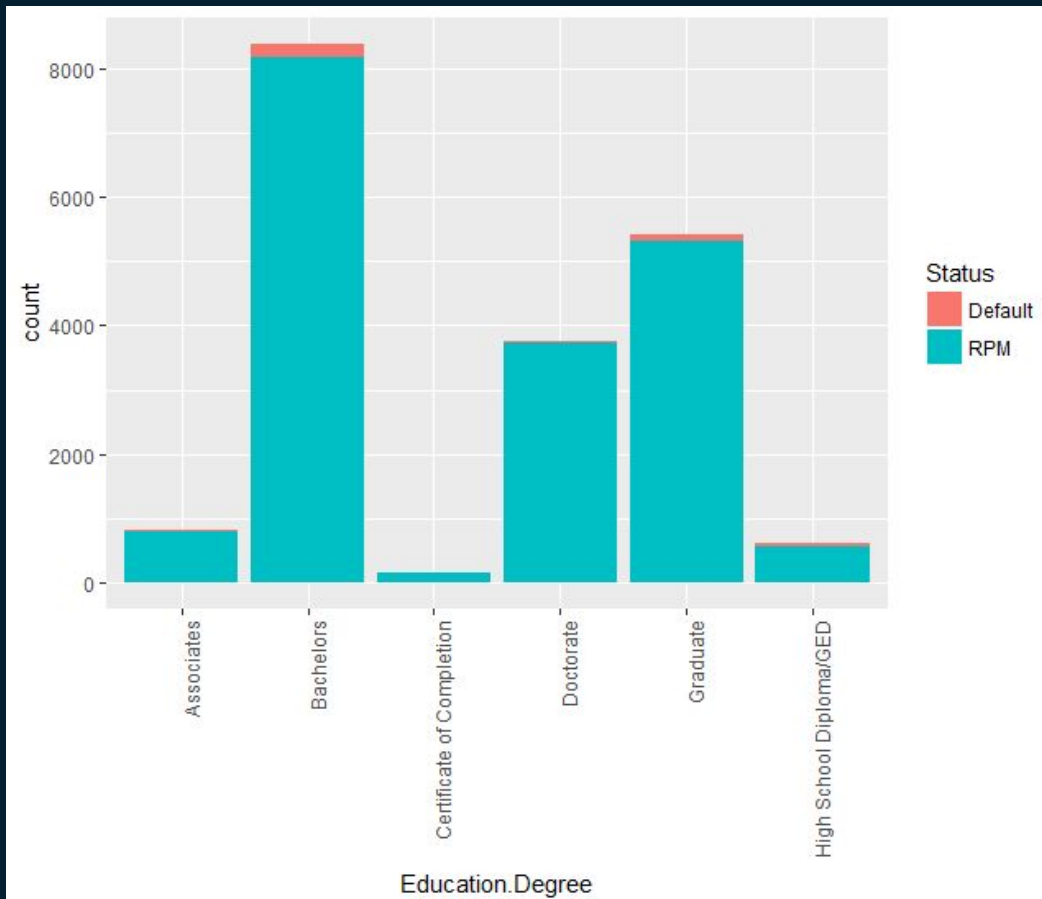
Exploratory Data Analysis

Loan disbursement dates of SLH App Users



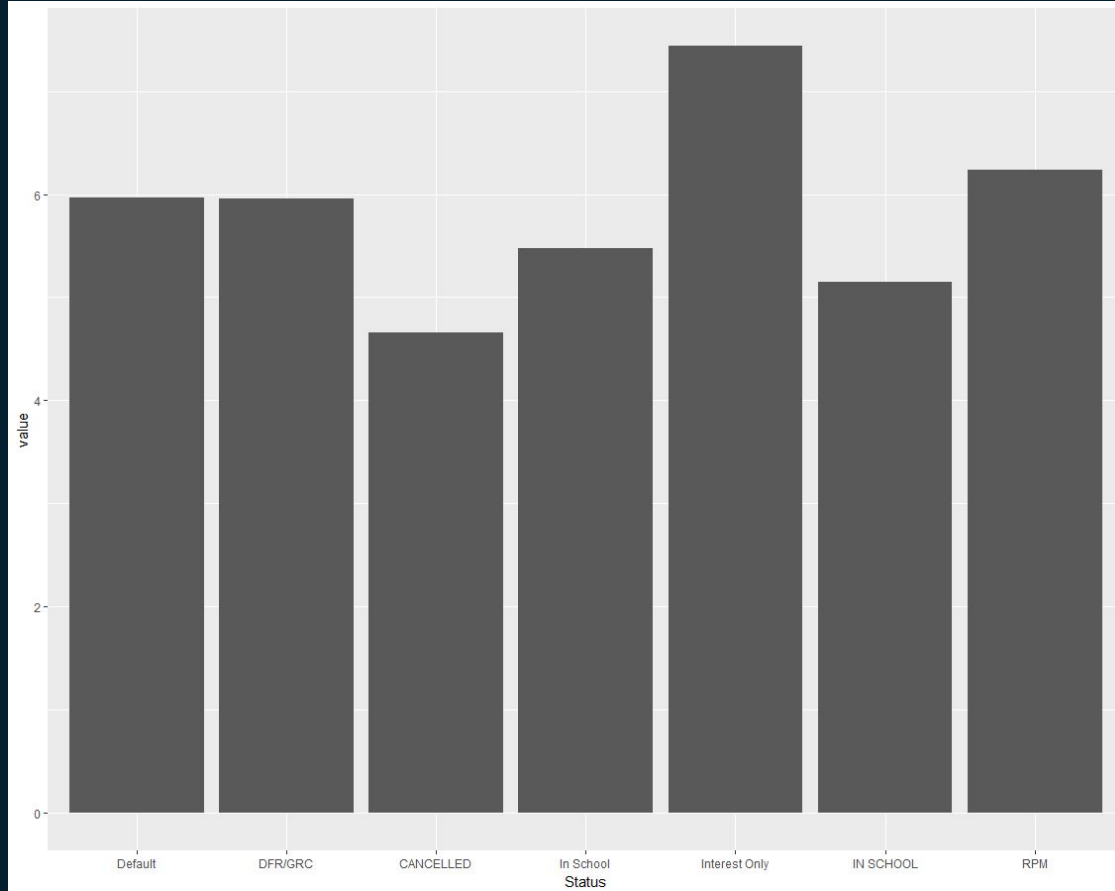
Exploratory Data Analysis

Loans Status by Degree Types



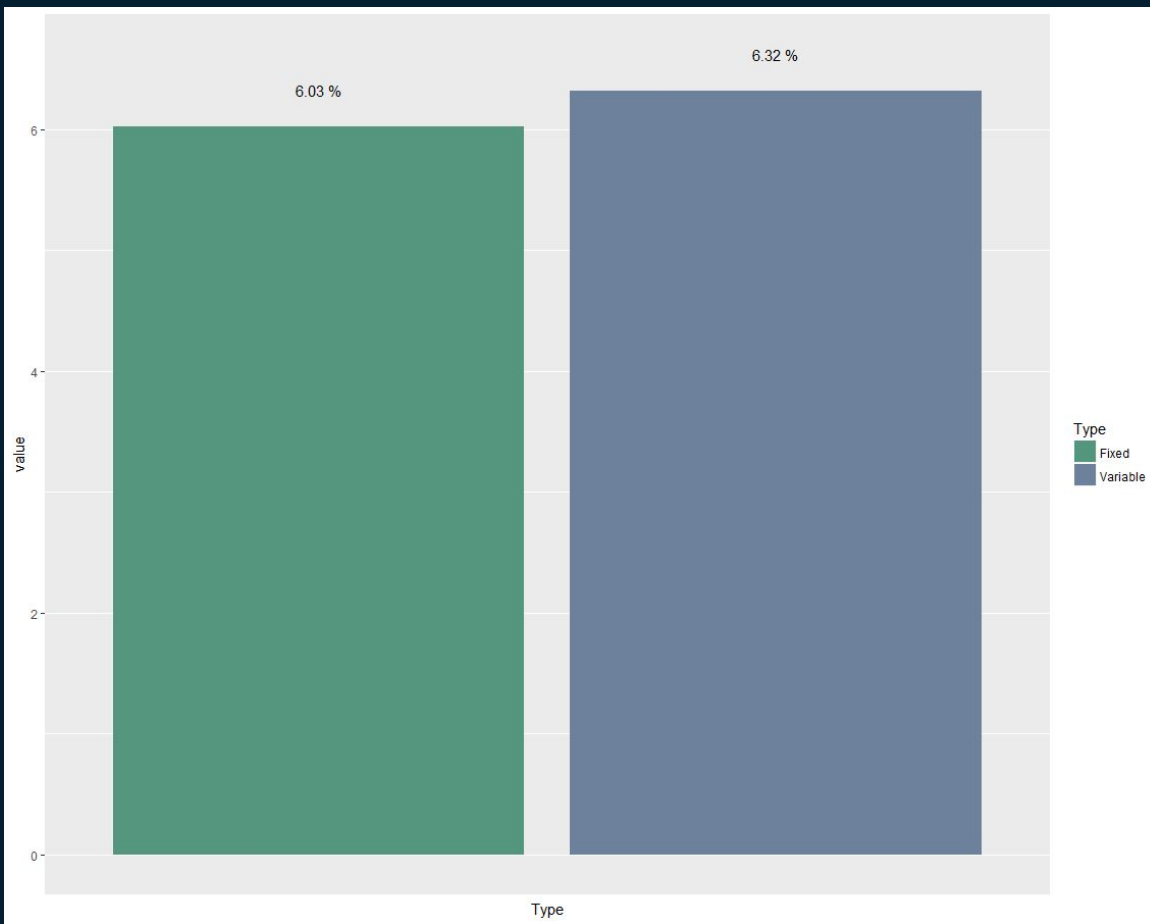
Exploratory Data Analysis

Average Interest Rate By Loan Status



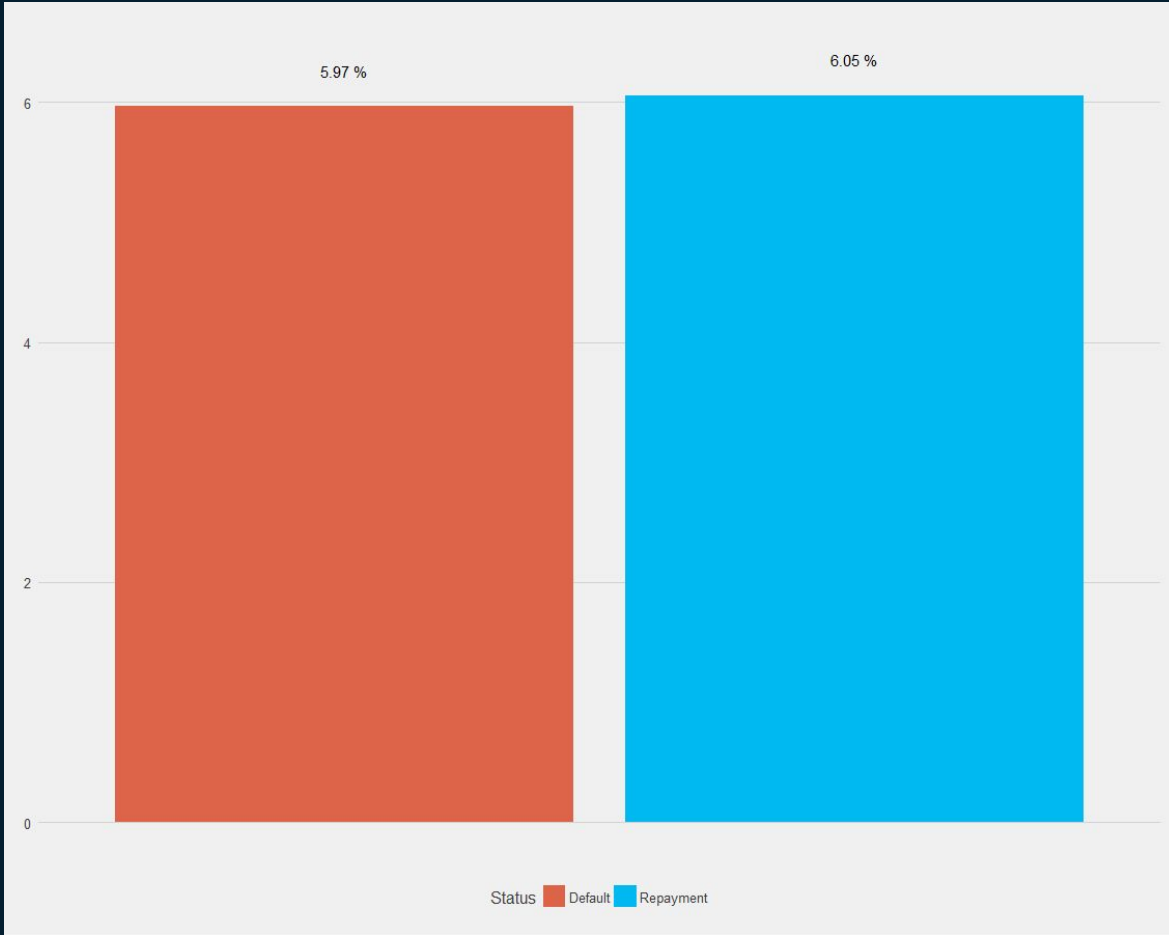
Exploratory Data Analysis

Average Interest Rates of Loans by Interest Rate Type



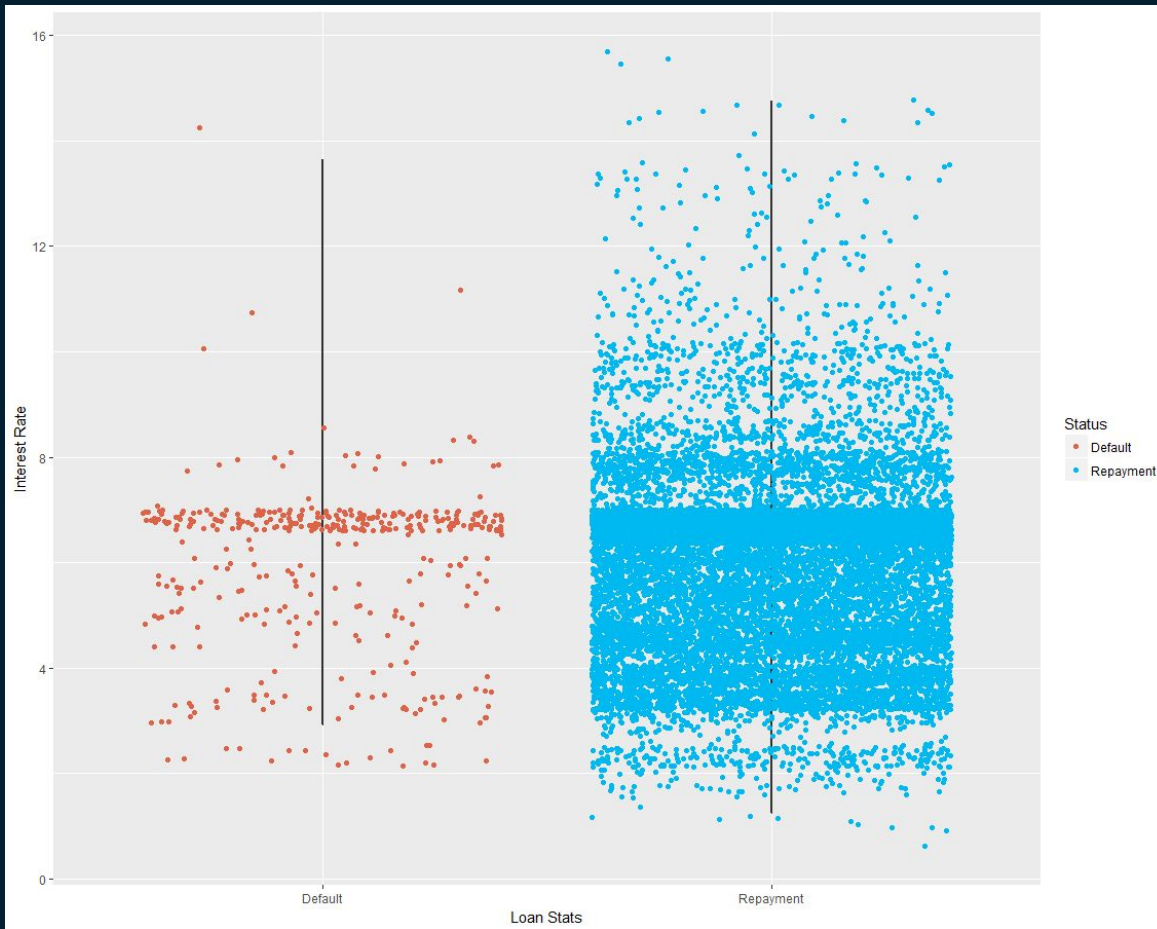
Exploratory Data Analysis

Average Interest Rates of Loans by Status



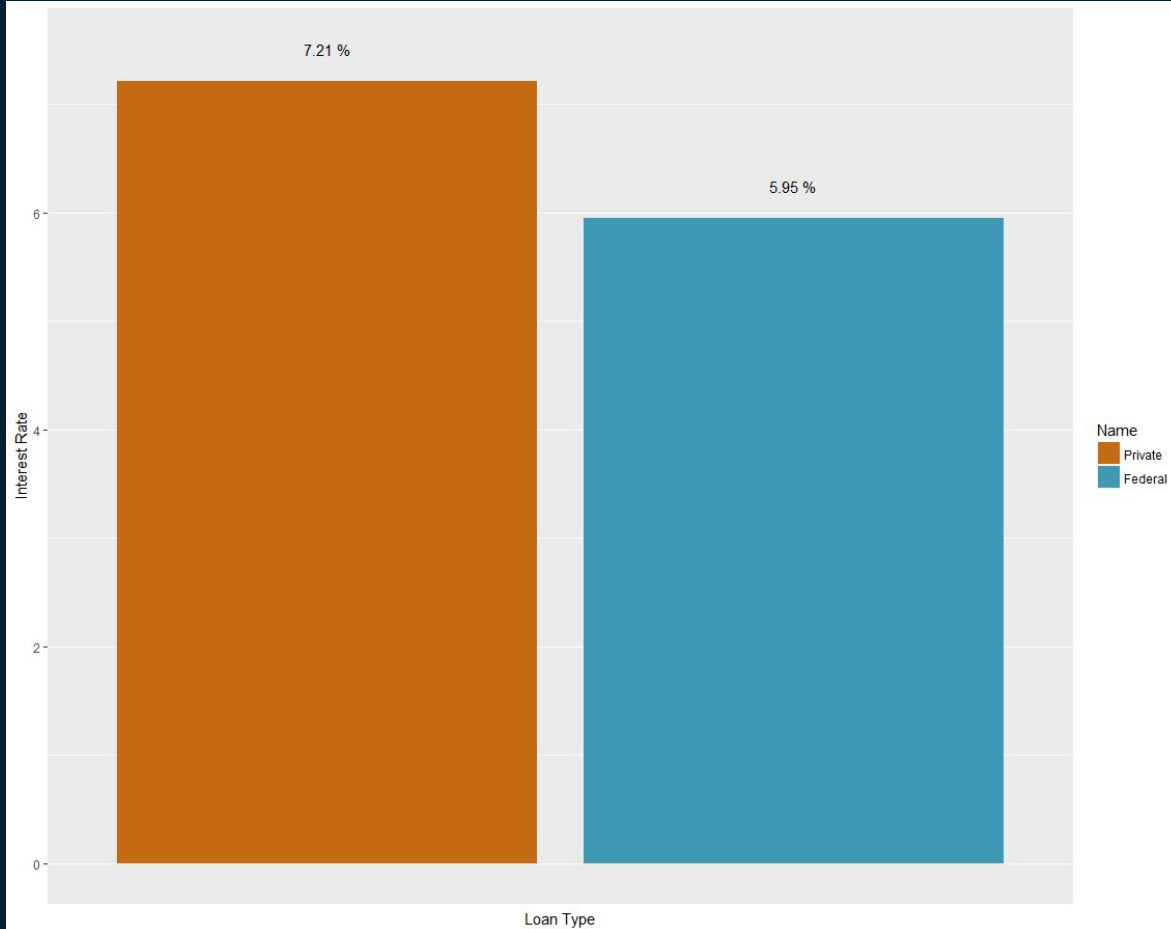
Exploratory Data Analysis

Average Interest Rates of Loans by Status



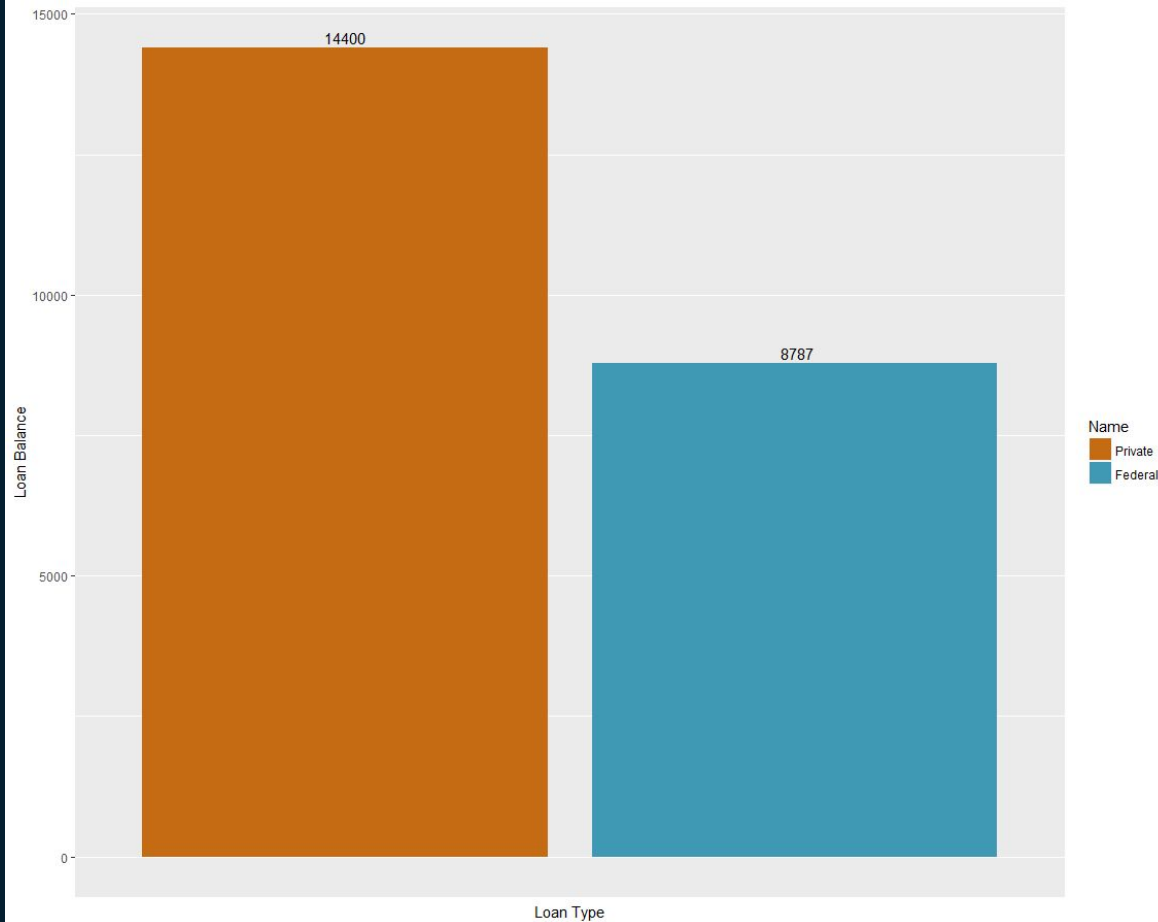
Exploratory Data Analysis

Average Interest Rates of Loans by Loan Type



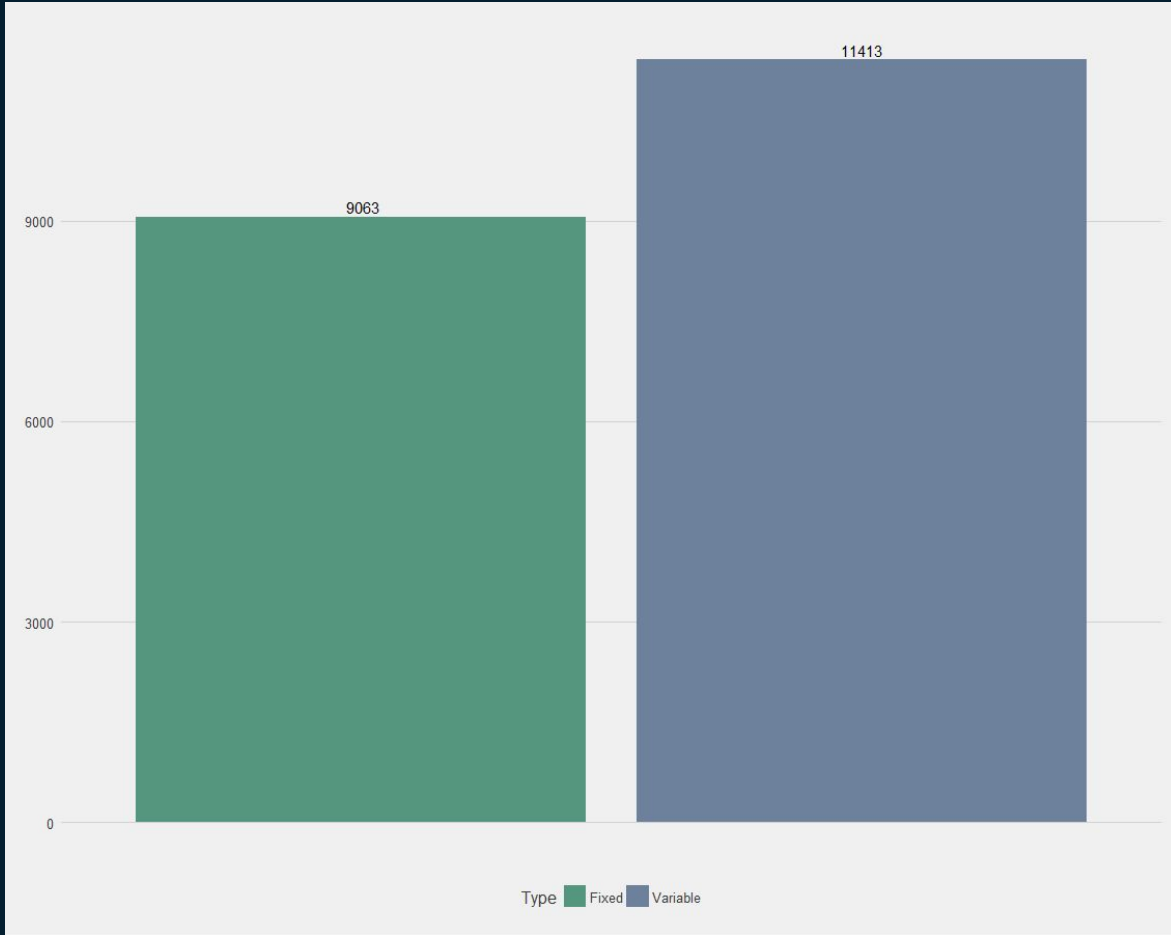
Exploratory Data Analysis

Average Balance of Loans by Loan Type



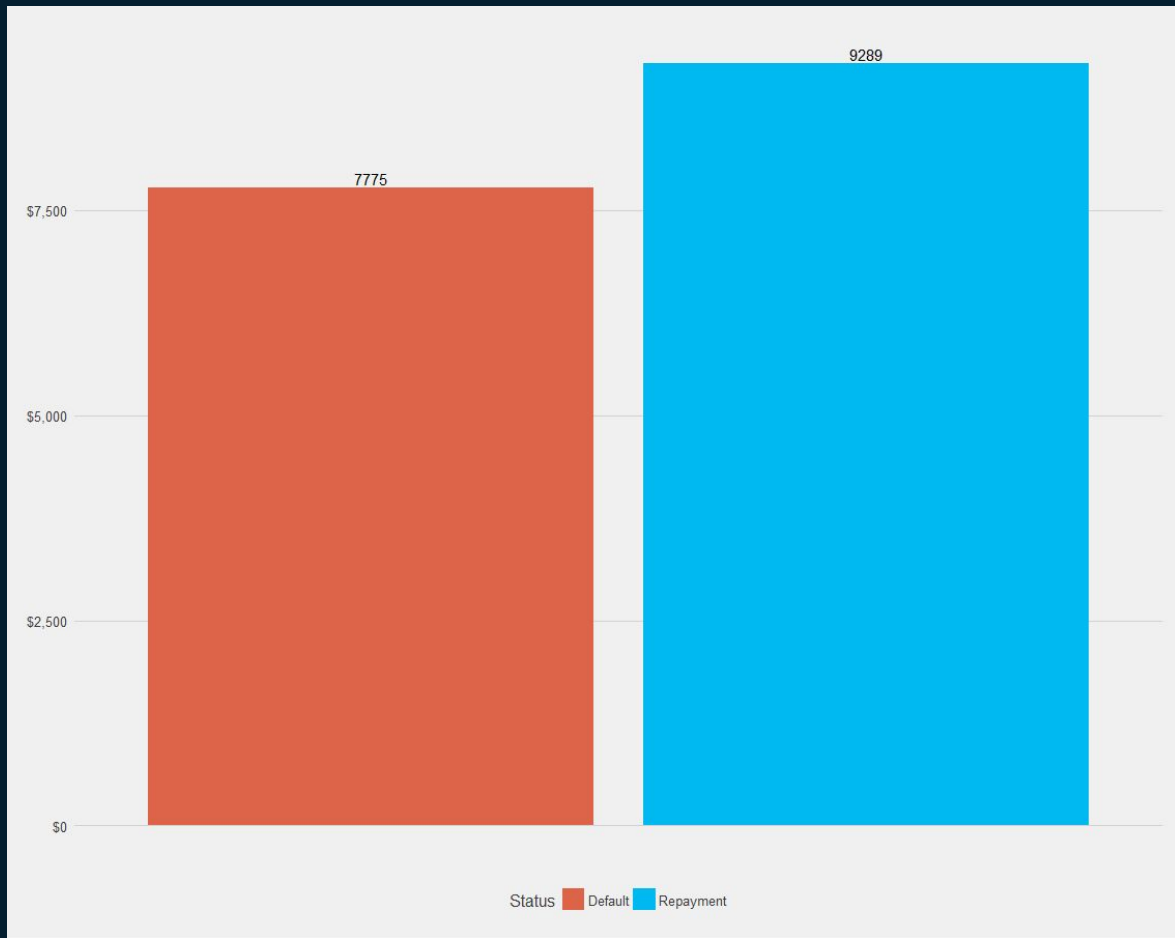
Exploratory Data Analysis

Average Balance of Loans by Interest Rate Type



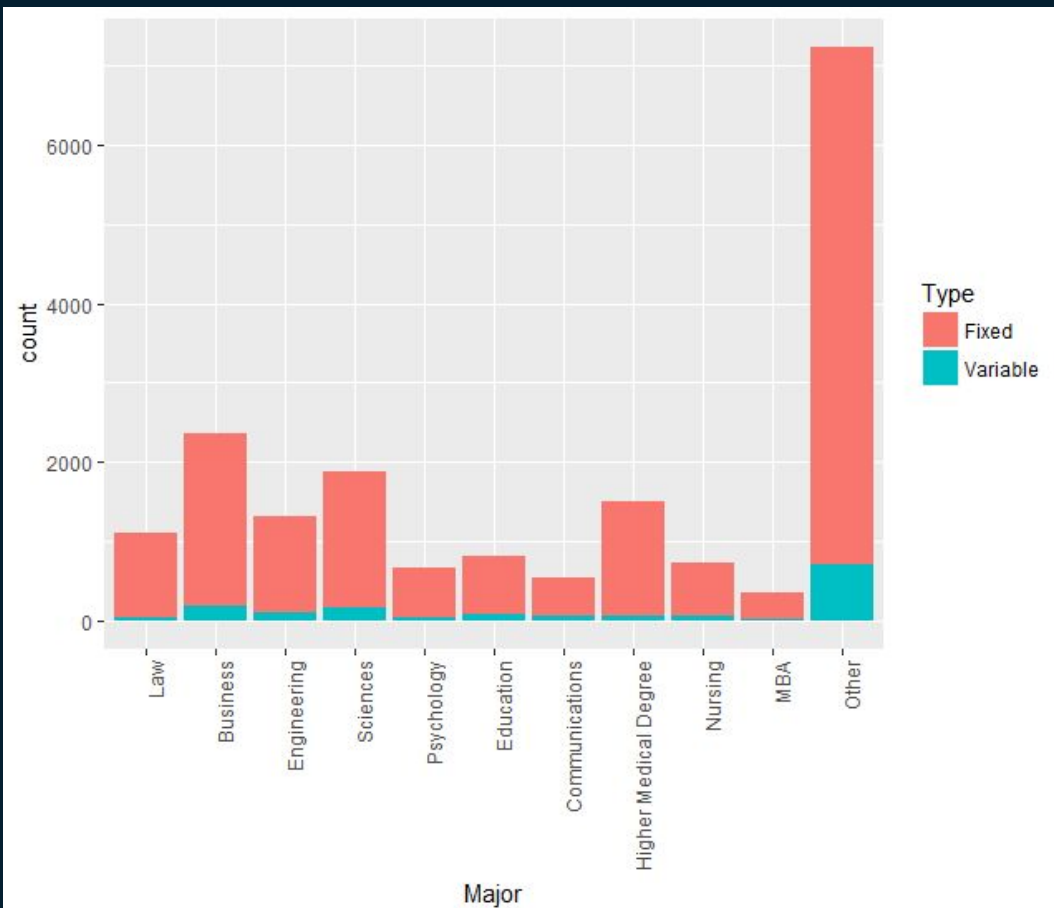
Exploratory Data Analysis

Average Balance of Loans by Interest Loan Status



Exploratory Data Analysis

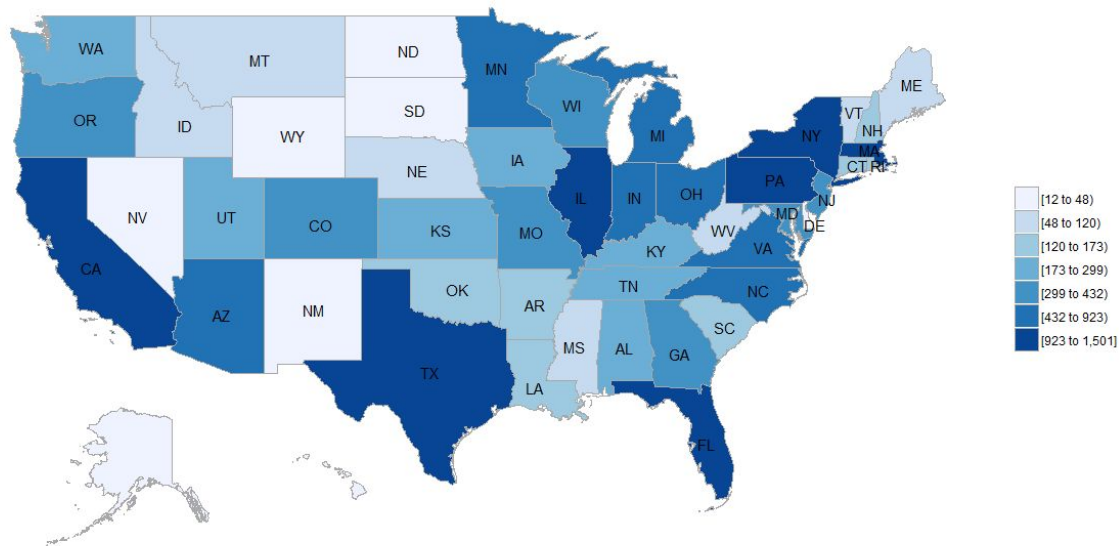
Count of total loans by interest rate type



Exploratory Data Analysis

Total Loans Per State

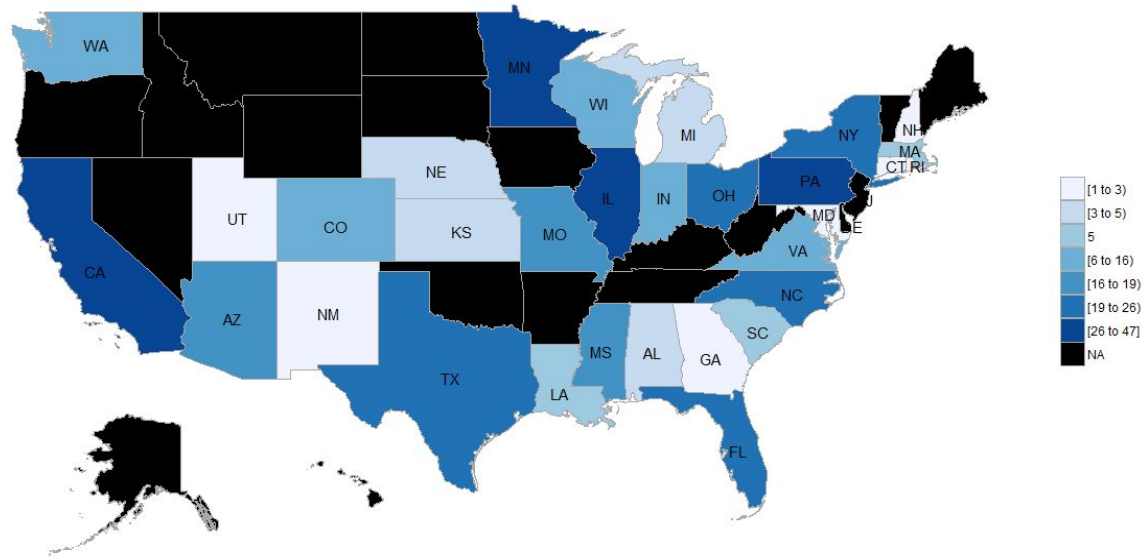
Loans Per State



Exploratory Data Analysis

Total Defaulted Loans per State

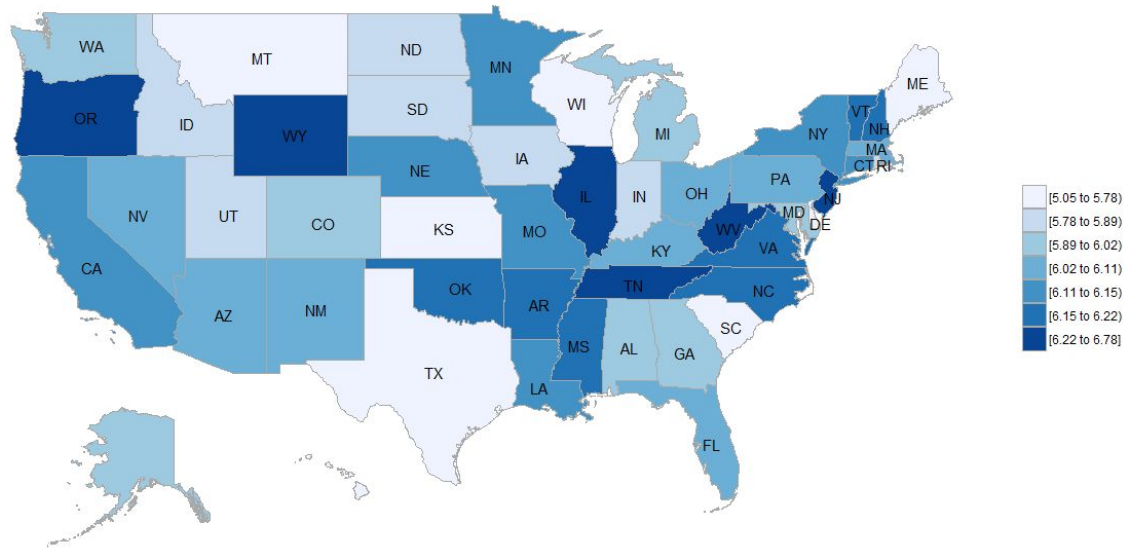
Loans As Per Status:Default



Exploratory Data Analysis

Average Loan Interest Rate by State

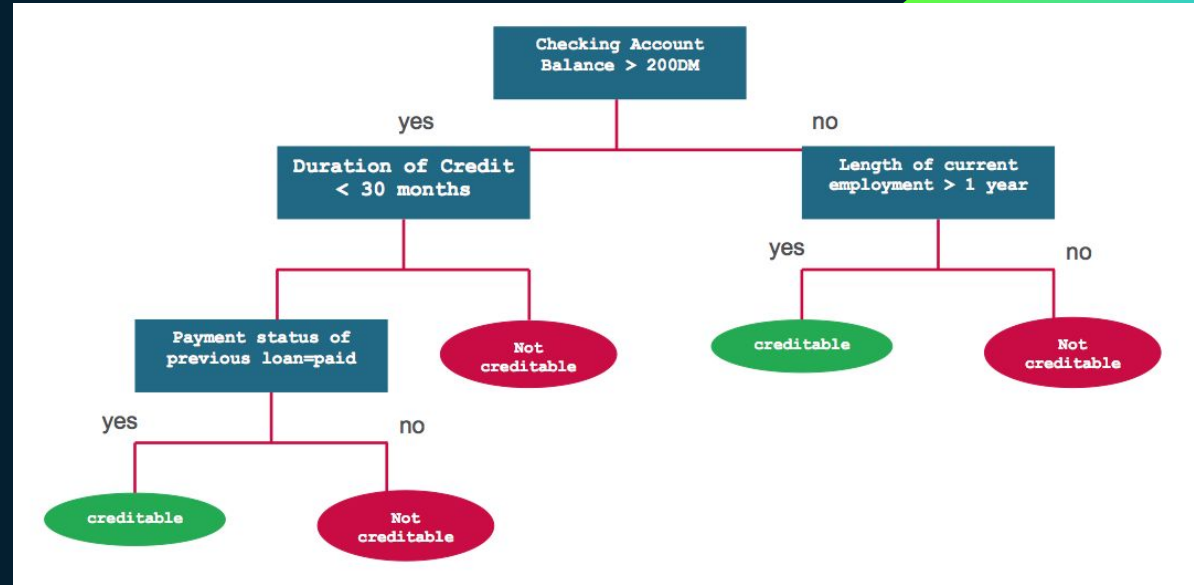
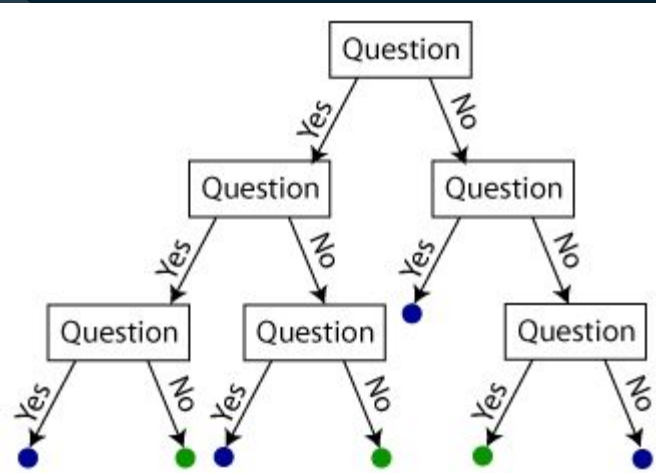
Average Rate of Interest



Predicting Student Loan Default Rates

Machine Learning Algorithm - Random Forest

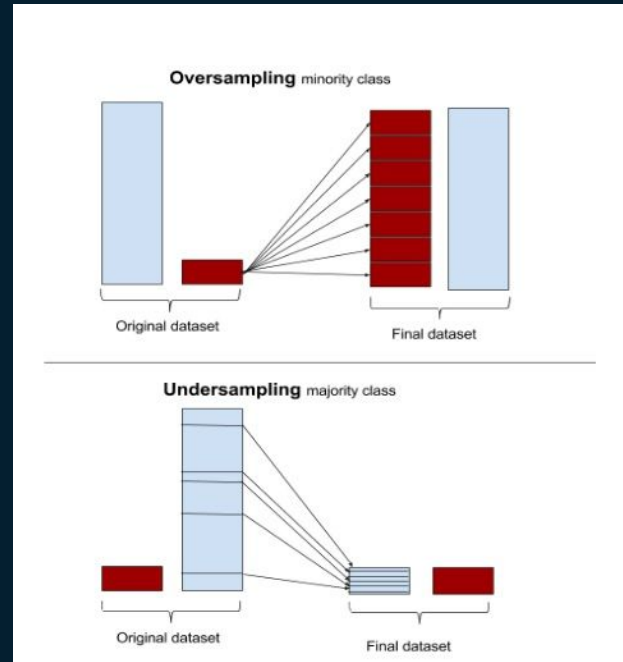
The goal of our prediction model was to determine whether the loan status of a given borrower would be "default" or "repayment". In determining the best machine learning algorithm to apply to our classifier model Random Forest was chosen for its accuracy, efficiency, methods available for balancing errors in class imbalance cases, and robust variable importance estimates.



Unbalanced Data Set - Class Imbalance

Undersampling vs Oversampling

The Student Loan Hero dataset is inherently unbalanced, with the vast majority of statuses being in repayment, and only a very small minority in default. In order to optimize our random forest algorithm to be as accurate as possible, it was decided to undersample the majority class, or the instances of statuses in repayment. This balanced the frequencies of default to repayment more closely.



Random Forest Model

Training the Model

Once undersampled, the dataset was split in order to train and then test the model. I settled on a 70/30 split of training to testing for the dataset.

```
loans1 <- dplyr::select(loans_under_balanced, -V1,-User.ID.,-X,-User.DOB,-Loan.ID.,-College,
-Joint.Federal.Income.Tax.,-Joint.Federal.Income.Tax.,-univstate,-univtribal)

loans1$Family.Size <- as.numeric(loans1$Family.Size)
#Divide into training and test datasets
set.seed(666)
split <- base::sample(nrow(loans1), floor(0.7*nrow(loans1)))
train <- loans1[split,]
test <- loans1[-split,]
```

The random forest model was then run on the training data set.

```
model <- randomForest(Status ~ ., data=train,
                      ntree=500,
                      mtry=6,
                      importance=TRUE,
                      na.action = na.roughfix,
                      replace=FALSE)
```

```
model
```

Random Forest Model

Random Forest Model Output

The results of running the model on the training dataset were as follows:

```
call:
  randomForest(formula = Status ~ ., data = train, ntree = 500,      mtry
= 6, importance = TRUE, replace = FALSE, na.action = na.roughfix)
      Type of random forest: classification
      Number of trees: 500
No. of variables tried at each split: 6

      OOB estimate of  error rate: 5.71%
Confusion matrix:
      Default Repayment class.error
Default      25         3  0.10714286
Repayment     1        41  0.02380952
```

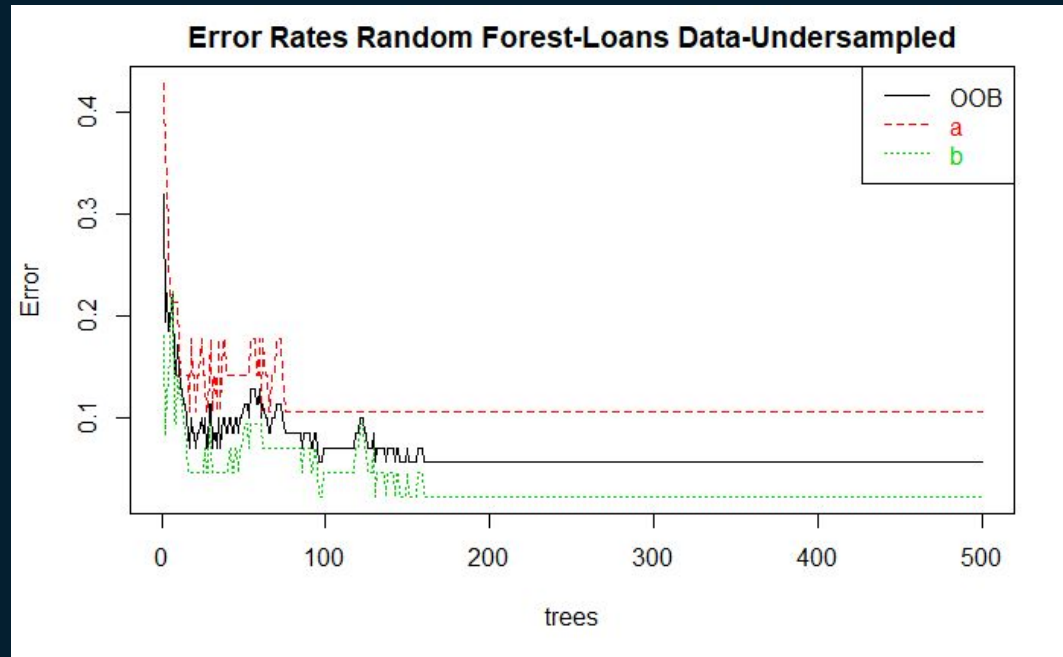
Our model has an accuracy of 89.3% in predicting when a loan is expected to be in default.

Our model has an accuracy of 97.6% in predicting when a loan is expected to be in repayment.

Random Forest Model

Random Forest Model - Out of Bag Error Rates

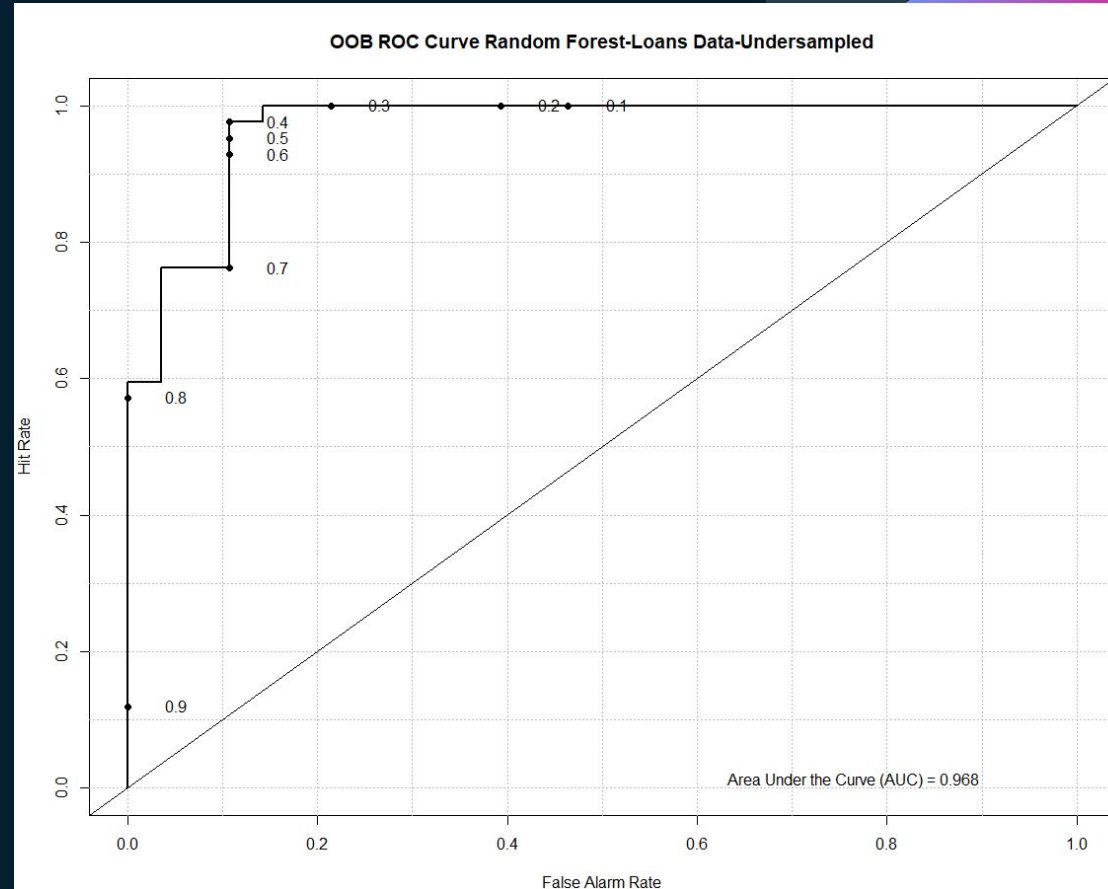
At a high level, the out-of-bag (OOB) error is the mean error rate in predicting the data left out of the training set for that tree.



Random Forest Model

Random Forest Model - Area Under the Curve

AUC is used as a metric to differentiate the prediction accuracy of the random forest model for loans in default and those in repayment. A value closer to 1 means that our model was able to correctly differentiate from a random sample of the two target classes of two loans in repayment and in default.



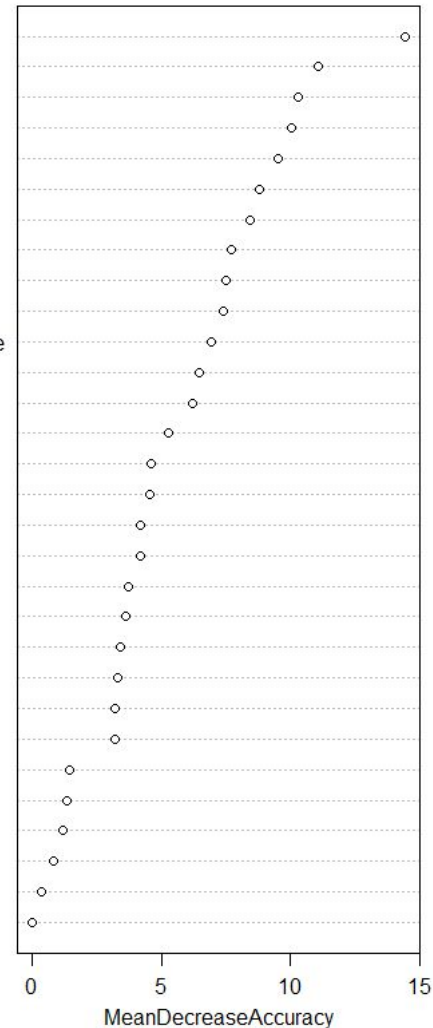
Random Forest Model

Variable Importance Map

The variable importance map shows how important each variable is in predicting whether a loan will be in default or will be in repayment.

The output for our random forest model shows that Credit Score is the most important variable in predicting whether a given loan is expected to be in default or in repayment.

Credit.Score
Major
univenrprofile
origprincipaluserbucket
usercurrentbal
origprincipaluser
Adjusted.Gross.Income
Employer.Type
univlocale
State.of.Residency
Spouse.Adjusted.Gross.Income
Rate
Profession
totalusercurrentbalbucket
Family.Size
univobereg
Original.Principal
univctrl
Difference
Monthly.Interest
Current.Principal
loancurrentbal
originalprinloan
Education.Degree
Employment.Status
Type
loancurrentbalbucket
originalprincipalloanbucket
univmedical
Name



Next Steps and Recommendations

Predicting Student Loan Default Rates

- With the basic model for predicting whether or not a user will find themselves in "default", Student Loan Hero could better target users at risk of defaulting on their student loans to deliver specific advice via email, phone, or direct mail, proactively monitor high-risk users, and better tailor in-app messages and advertising campaigns to better serve users based on their likelihood of default.
- Student Loan Hero could develop a proprietary score or grade that is assigned to each user based on the health of their profile according to our default model. Perhaps an A-F rating of how a user is doing on their debt payoff journey. This could be used as an user acquisition tool, or for press and outreach.
- With the amount of users and loans available in the constantly growing Student Loan Hero user database, Student Loan Hero could develop an underwriting model to share with refinancing partners in an effort to obtain optimized underwriting models for Student Loan Hero users at their refinancing partners, or simply to allow our partners to tweak their own underwriting criteria and models.