STOUT CASE STUDY

Name: **Atharva Saykhedkar**

Email: **atharvasaykhedkar@gmail.com**

Github Link: **https://github.com/atharv52/Stout-Case-Study**

Html link: **stout-case-study.tiiny.site**

The following document contains the problem statements and the elaborate procedure I followed to come up with solutions. All code is available on the github link. Please refer to the code while going through this document to get a clear idea about results.

For case study 1 please refer to casestudy.r file in the github repo.

For Case study 2 please refer to casestudy2.ipynb or casestudy2.py file.

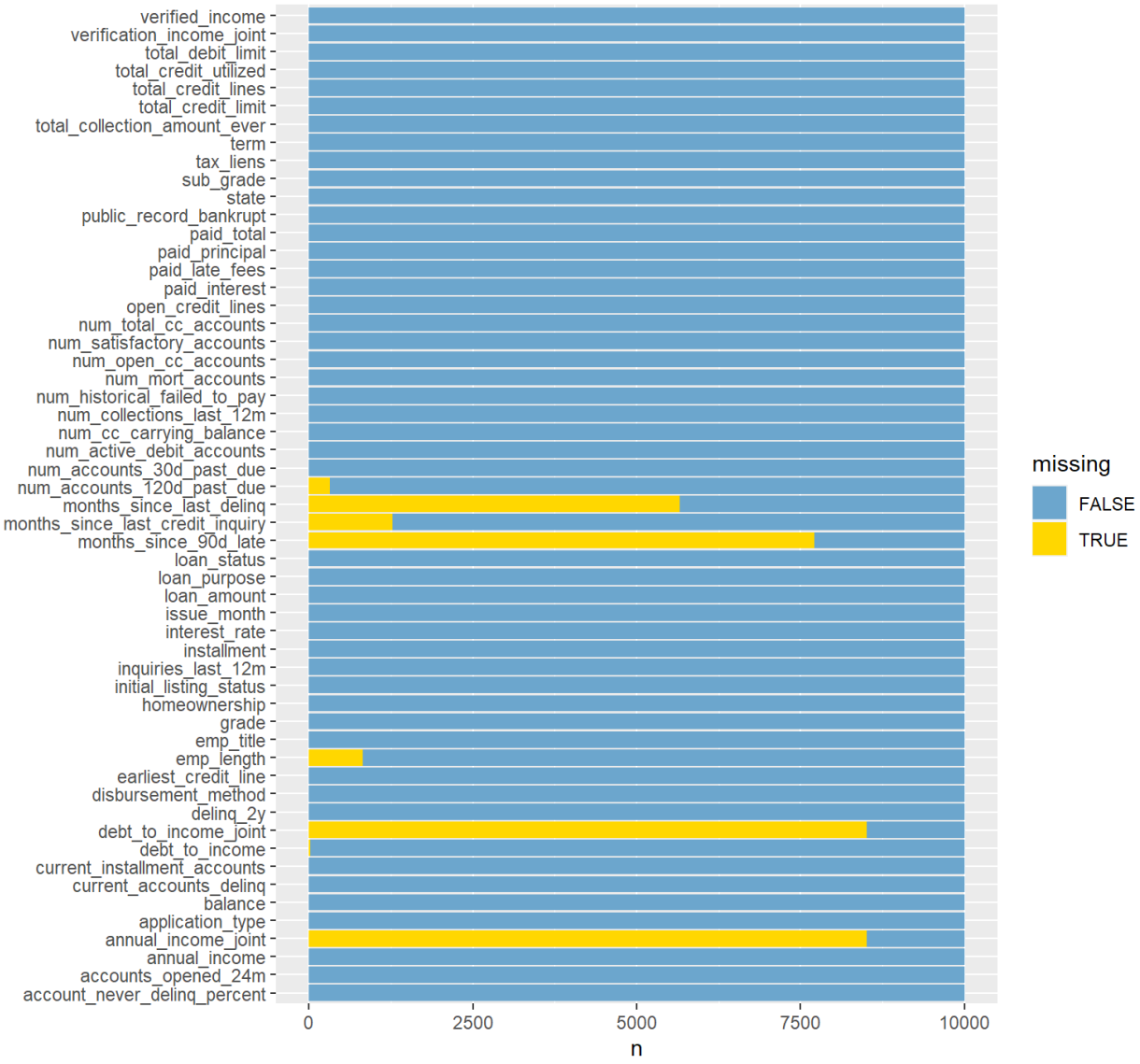
Case Study #1

Below is a data set that represents thousands of loans made through the Lending Club platform, which is a platform that allows individuals to lend to other individuals.

We would like you to perform the following using the language of your choice:

- Describe the dataset and any issues with it.

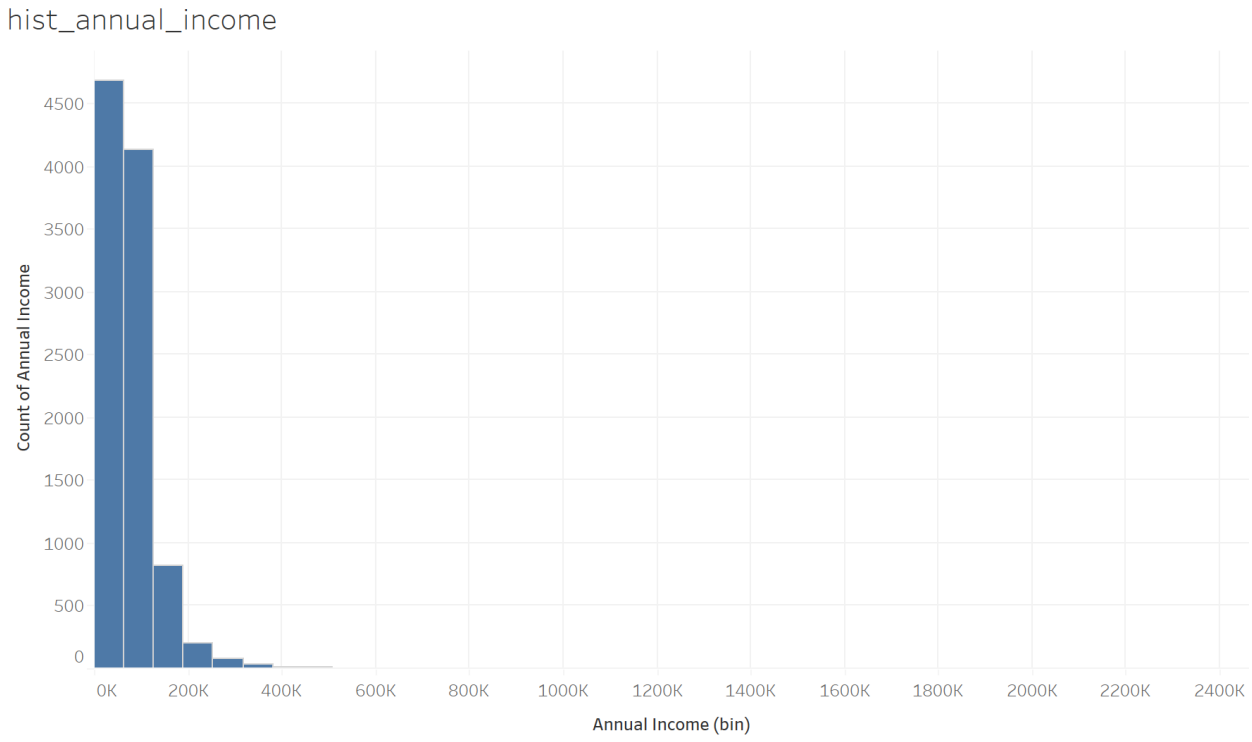
  The dataset is about the loans that are made through the lending club platform and all the different information associated with it.
  Some of the issues that were found in the dataset:
  - The number of NA values in various columns. The following image shows the count of missing values in various columns



- Another issue with na values is that they are in character type and not as type N/A.
- For Joint loans there is no employee title and no employee length.
- Some of the employee titles have abbreviations which are not clearly defined as well as vague employee titles For eg: Deputy.
- For Joint loans there is annual income present as well as annual_joint_income present. For this I feel only joint annual income would've been suffice as it might create confusion while performing analysis because a joint loan should be assessed only over joint annual income and not annual income and presence of both create confusion.
- As this dataset is about loans actually made, we can say that we have data about customers who have mediocre to good credit history as their applications were accepted. Also there might be some exceptions where a person having relatively poor credit history was offered a loan with high interest rate and the accepted it out of desperation. Therefore while predicting the interest rate column we can say that

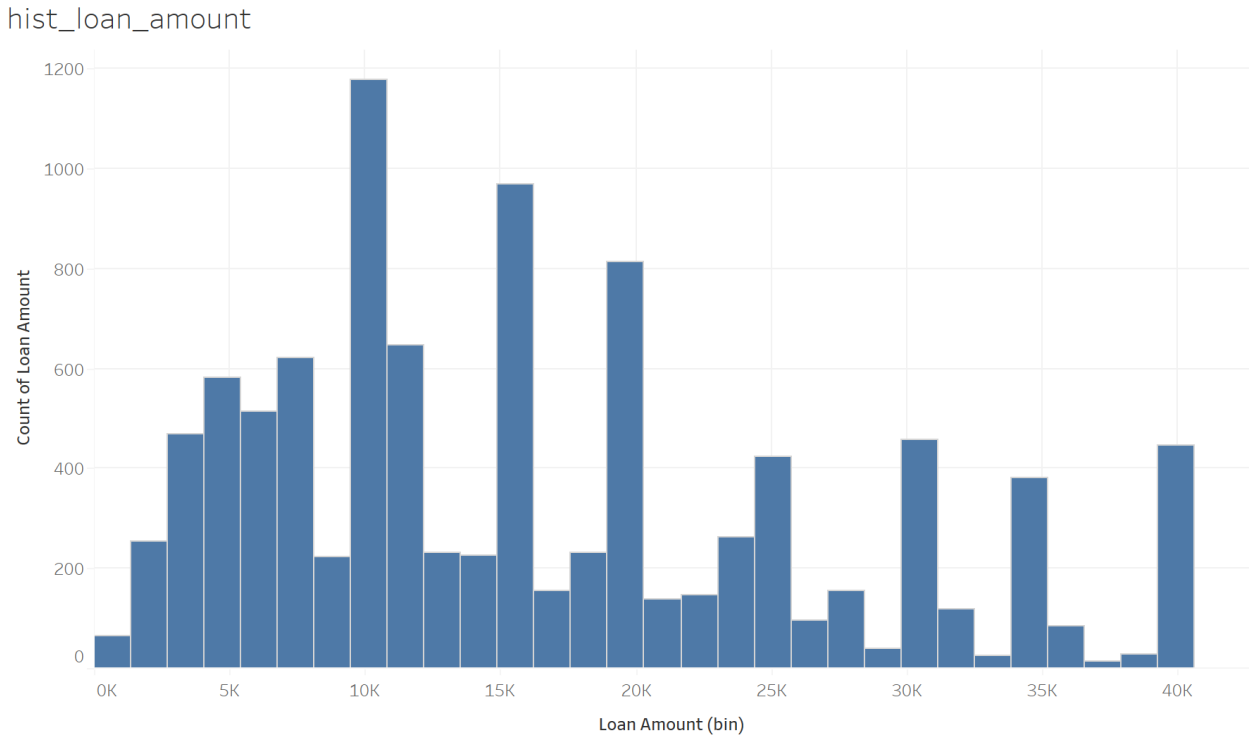
we don’t have enough data about customers who were assessed to not pay the loan back and were charged very high interest rate or their application was rejected. This might have some effect on the predictions.

- Generate a minimum of 5 unique visualizations using the data and write a brief description of your observations. Additionally, all attempts should be made to make the visualizations visually appealing
    - For visually appealing visuals the following histograms were produced in Tableau.
    - Histogram of annual income shows how the data is spread out. Around half of the records have a income between 0k and 75k.

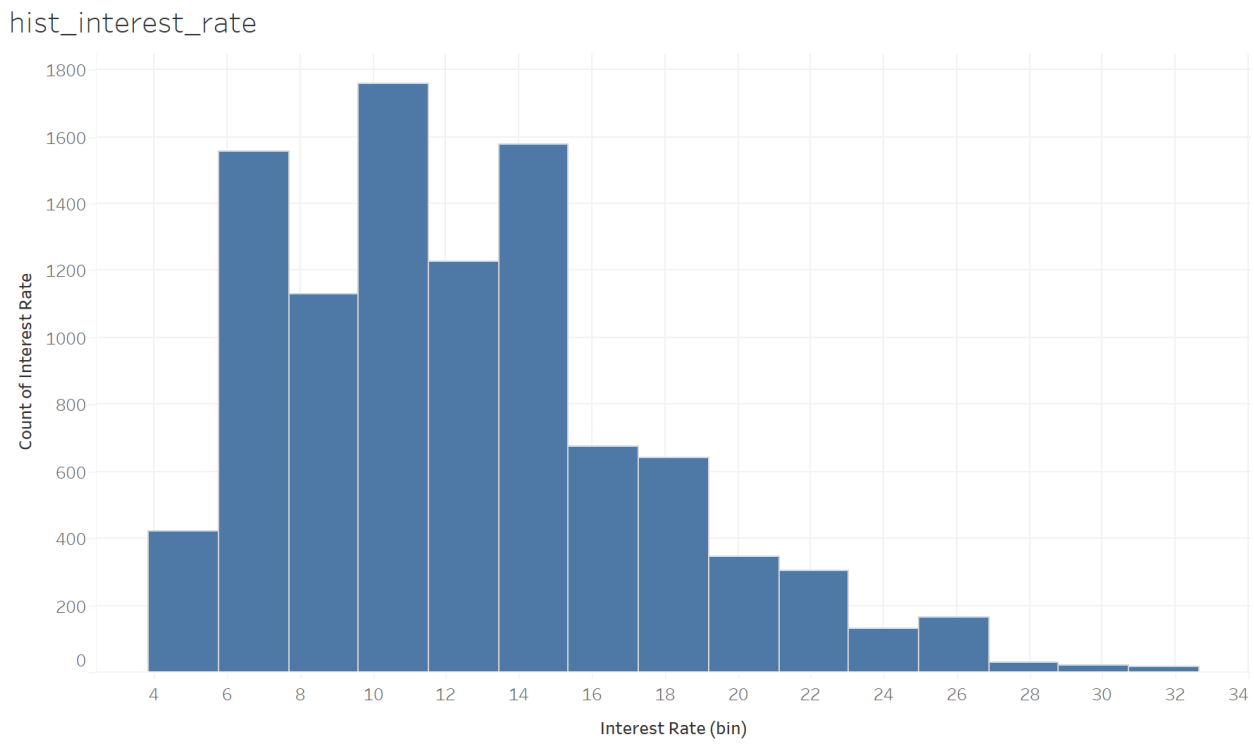


The trend of count of Annual Income for Annual Income (bin).

- Histogram of loan amount. We can see how the loan amount is distribute among the records. Where 40k is the highest loan amount.

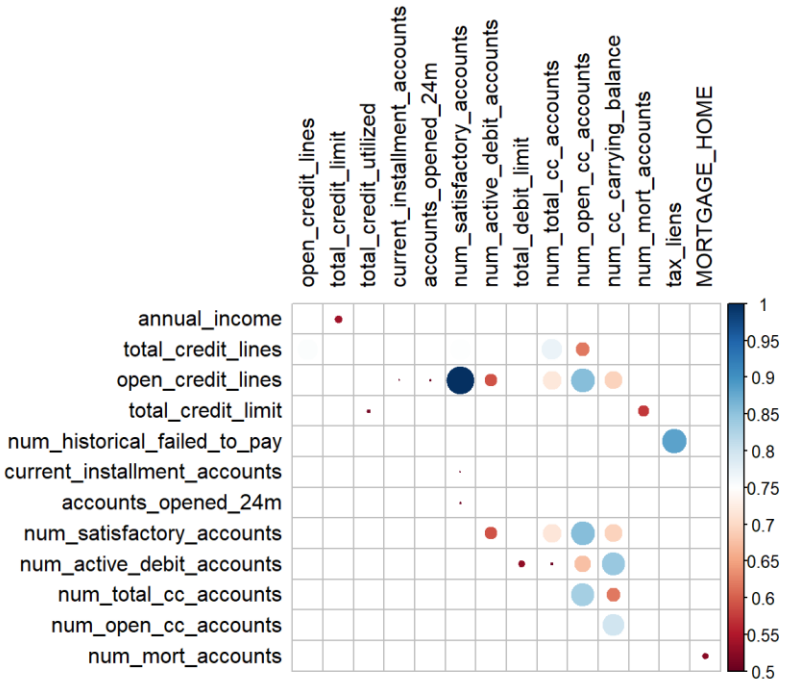


The trend of count of Loan Amount for Loan Amount (bin).



The trend of count of Interest Rate for Interest Rate (bin).

- Histogram of interest rate where we can see how the interest rate is distributed. We can see there are a some records having an interest rate of more than 30%.

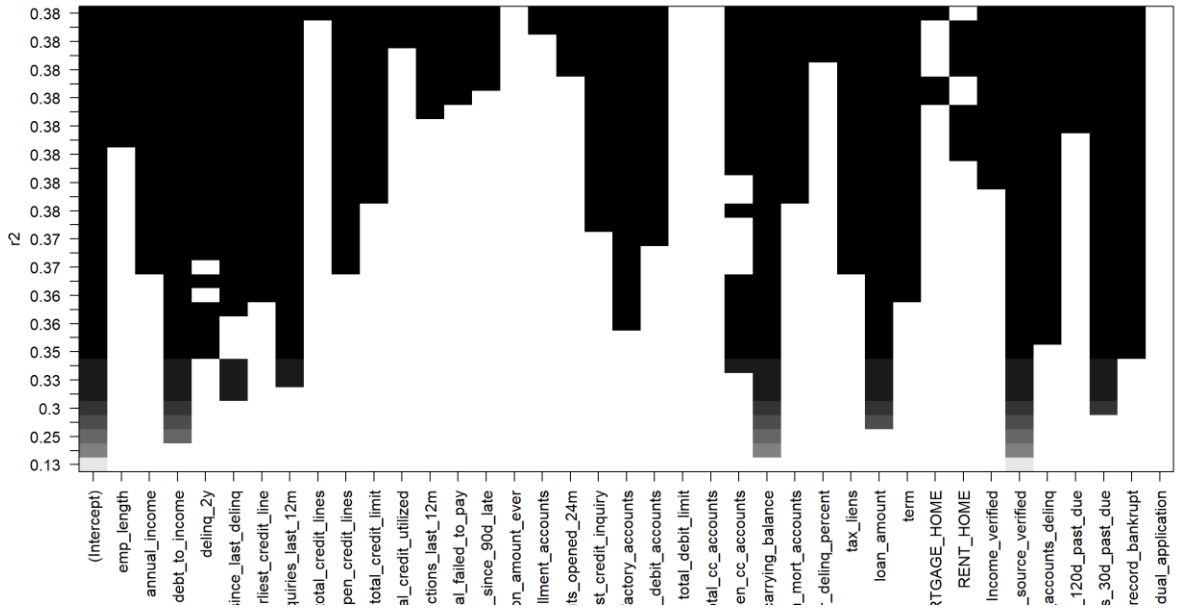


- The above correlation matrix was created in R to keep only the highest correlation between variables to reduce noise.

- Create a feature set and create a model which predicts *interest_rate* using at least 2 algorithms. Describe any data cleansing that must be performed and analysis when examining the data.
    - First, we remove character columns like emp_title, state, grade, sub-grade, issue_month and loan purpose.
    - Also removing some of the columns which are dependent on the interest rate or which will be defined after determining the interest rate like installment, loan_status, initial_listing_status, disbursement_method, balance, paid_total, paid_principal, paid_interest, paid_late_fees
    - After that we will encode the categorical variables of homeownership, verified_income, verification income joint, application_type
    - We look for outliers in the income and debt to income columns as this might affect the accuracy of the model.
    - We will split the datset into train test and validation split.

    TO CREATE FEATURESET:

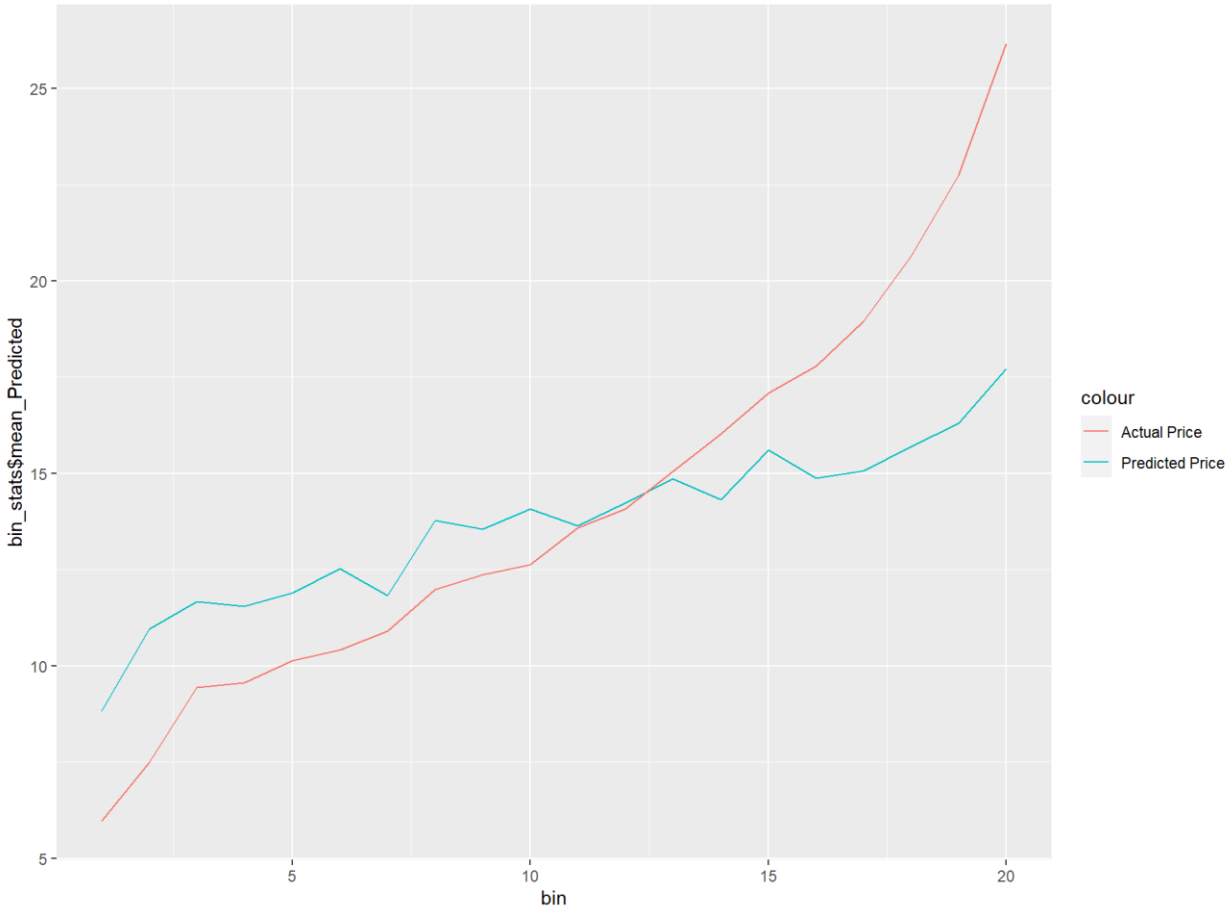
    - We can use regsubsets to see the maximum R squared value at each iteration for each variable. We will use methods such as backward elimination and forward selection to find out the most important variables. We will select some of the most important variables to minimize overfitting.
    - Assuming that joint applications will require that we assess the joint income and individual applications be assessed over individual income I feel the joint loans records made should be removed from the dataset as it will affect the individual loans made while predicting interest rate. Moreover the fields like total credit lines, total accounts, months since last inquiry etc are for multiple people for a single person in joint loans is not mentioned. Hence it will be better to have a separate dataset of joint loans made to predict their interest rates. Also there are a lot of NA values in the joint income column which will introduce error while predicting.



- Looking at the above graph we can see that the following are the most important features that explain the interest rate at each iteration.
    - Income Source verified, Num_cc_carrying_balance, debt_to_income_ratio, loan_amount, num_accounts_30d_past_due
- After doing linear regression by selecting features by backward elimination by AIC we get an adjusted R2 of 0.36 also after making the predictions on the validation dataset we get a RMSE value of 3.9

- Visualize the test results and propose enhancements to the model, what would you do if you had more time. Also describe assumptions you made and your approach.
    - I would do cross validation on the dataset by creating more datasets.
    - Also would try to get data by oversampling the kind of records that we have less and undersampling the kind of records we have more

    Linear Regression Results:



- The above graph gives the difference between mean actual vs mean predicted values. We can see that the model does not do a good enough job at predicting the interest rate. We can say that only 36% of the variability in the interest rate can be explained by the independent variables.

REGULARIZATION (LASSO REGRESSION)

- After using the linear regression model we can see that the R squared value is not significant. Probably this maybe due to human behaviour and the model fails to account for it.
- We can use regularization such as lasso regression to minimize the noise in the predictor variables. Lasso Regression has the ability to minimize the coefficient of insignificant variables to zero. So this is an advance type of variable selection. On top of that lasso regression introduces a error variable lambda that will try to put a penalty on the MSE so that to reduce the sensitivity of interest rate on some of the variables.
- We create data matrix for the dataframe and predict the best lambda using k-fold crossvalidation. Using the value for lambda we fit the lasso regression model. We can use this model to predict the independent variable. By predicting the interest

rate on train data we got a R squared value of 37.7% and doing the same on validation data we got a R squared value of 36.6%.

- We can see the coefficients which lasso regression has reduced to zero.

Dataset

https://www.openintro.org/data/index.php?data=loans_full_schema

Output

An HTML website hosting all visualizations and documenting all visualizations and descriptions. All code hosted on GitHub for viewing. Please provide URL's to both the output and the GitHub repo.

* If you submit a jupyter notebook, also submit the accompanying python file. You may use python(.py), R, and RMD(knit to HTML) files. Other languages are acceptable as well.

Case Study #2

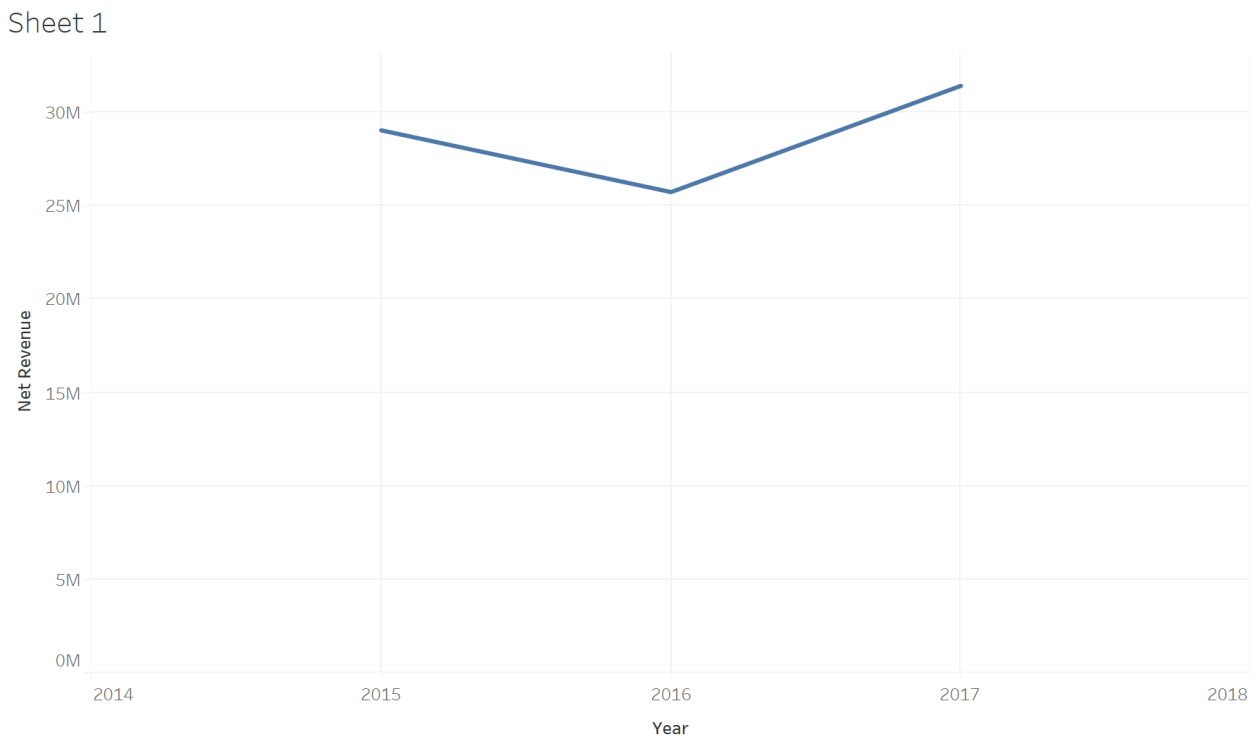
There is 1 dataset(csv) with 3 years worth of customer

orders. There are 4 columns in the csv dataset: index, CUSTOMER_EMAIL(unique identifier as hash), Net_Revenue, and Year.

For each year we need the following information:

- Total revenue for the current year
    - 31417495.030000016 (for year 2017)
- New Customer Revenue **e.g. new customers not present in previous year only**
    - 28776235.039999995 (New customers in 2017 Revenue)
- Existing Customer Growth. To calculate this, use the Revenue of existing customers for current year –(minus) Revenue of existing customers from the previous year
    - -4744565.19
- Revenue lost from attrition
    - 4744565.19
- Existing Customer Revenue Current Year
    - 2740887.39 (2017)
- Existing Customer Revenue Prior Year
    - 7485452.58 (2016)
- Total Customers Current Year
    - 249987 (2017)
- Total Customers Previous Year
    - 204646 (2016)
- New Customers
    - 229028 (2017)
- Lost Customers
    - 183687 (2016)

Additionally, generate a few unique plots highlighting some information from the dataset. Are there any interesting observations?



The trend of sum of Net Revenue for Year.

Above figure shows the revenue for the three years.

Dataset

https://www.dropbox.com/sh/xhy2fzjdvg3ykhy/AADAVKH9tgD_dWh6TZtOd34ia?dl=0

customer_orders.csv

Output

An HTML website with the results of the data. Please highlight which year the calculations are for. All code should be hosted on GitHub for viewing. Please provide URL's to both the output and the GitHub repo.

* If you submit a jupyter notebook, also submit the accompanying python file. You may use python(.py), R, and RMD(knit to HTML) files. Other languages are acceptable as well.