Which Loans Might Default? Lending Club Data Analysis and Prediction

Meskerem Goshime

September 22, 2022

Data Wrangling and Data Cleaning

➔ I started with loan data from 2007-2015 with 74 columns and 759,339 rows.

➔ Columns with 50k or more missing values were dropped.

➔ Null values were imputed with 0 or ‘Other’ where it made sense.

➔ After that, remaining rows with missing values were dropped.

➔ Took care of inconsistency in data entry by combining similar values in some columns.

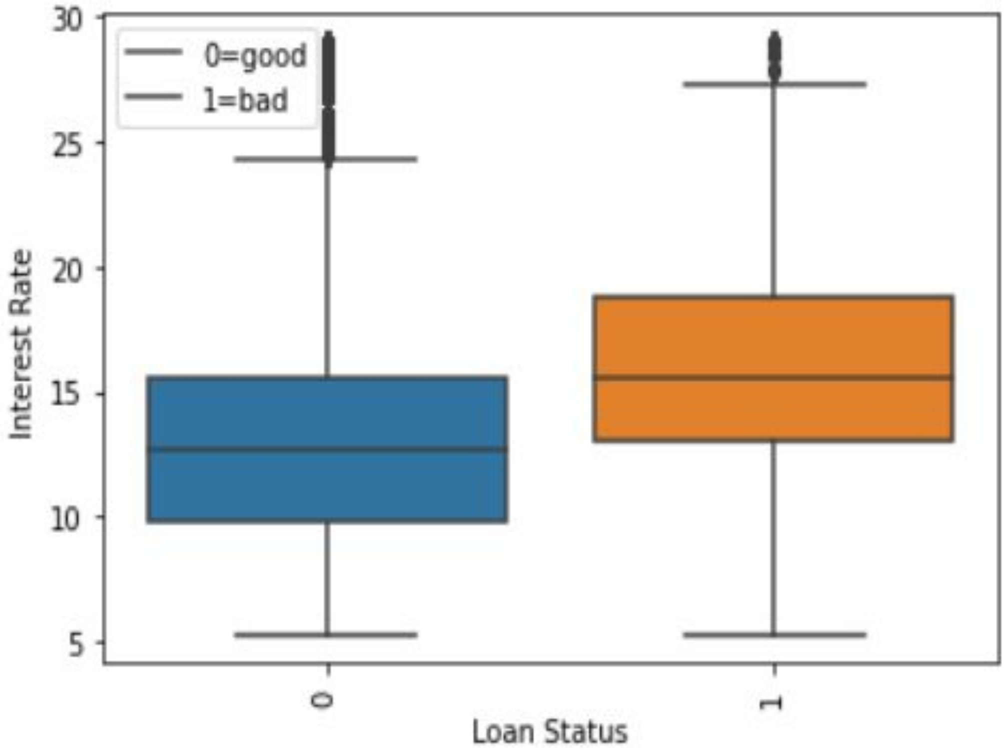
Outliers Handling

- ➔ The most significant outliers were in annual_inc and dti columns.
- ➔ Max Annual income = 9 million, 99.7 percentile 379 k
- ➔ Max dti = 380, 99.7 percentile 39
- ➔ Rows with values beyond the 99.7 percentile in the respective columns were removed.

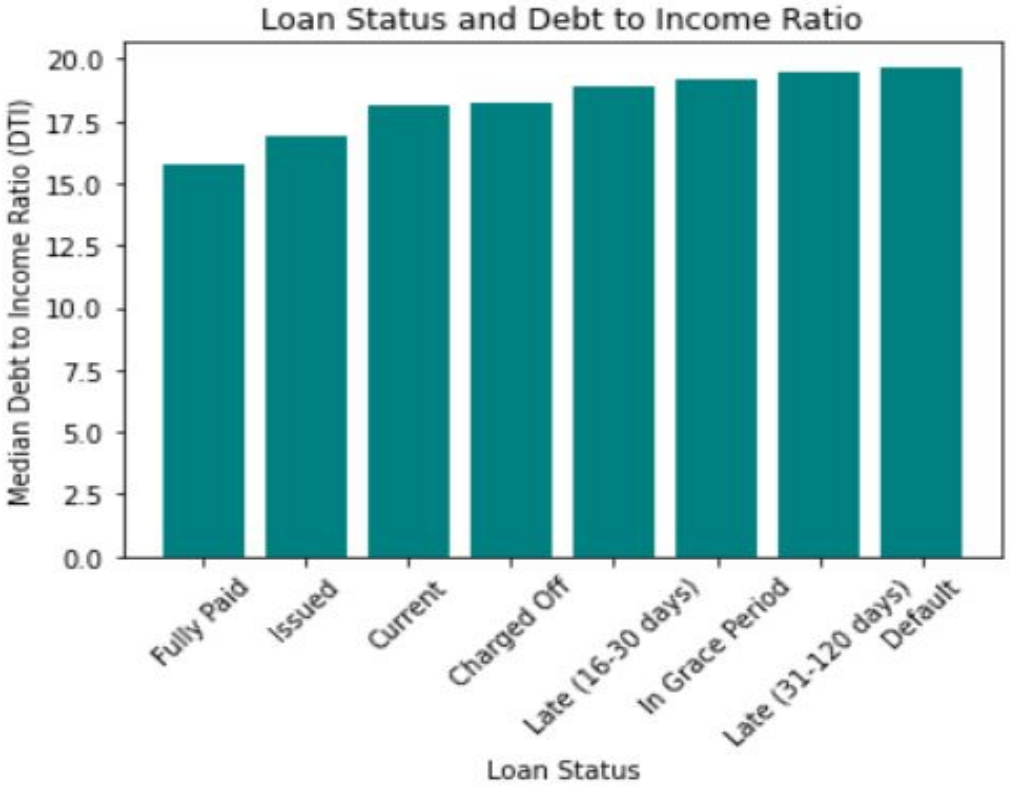
Feature Selection

- Columns with redundant data were dropped.
- Columns that are highly correlated were dropped.
- Dropped columns that seem unnecessary based on my intuition from the data exploration
- I ended up with 31 columns from 74.
- In the end I selected 10 columns out of the 31 using Feature Importance.

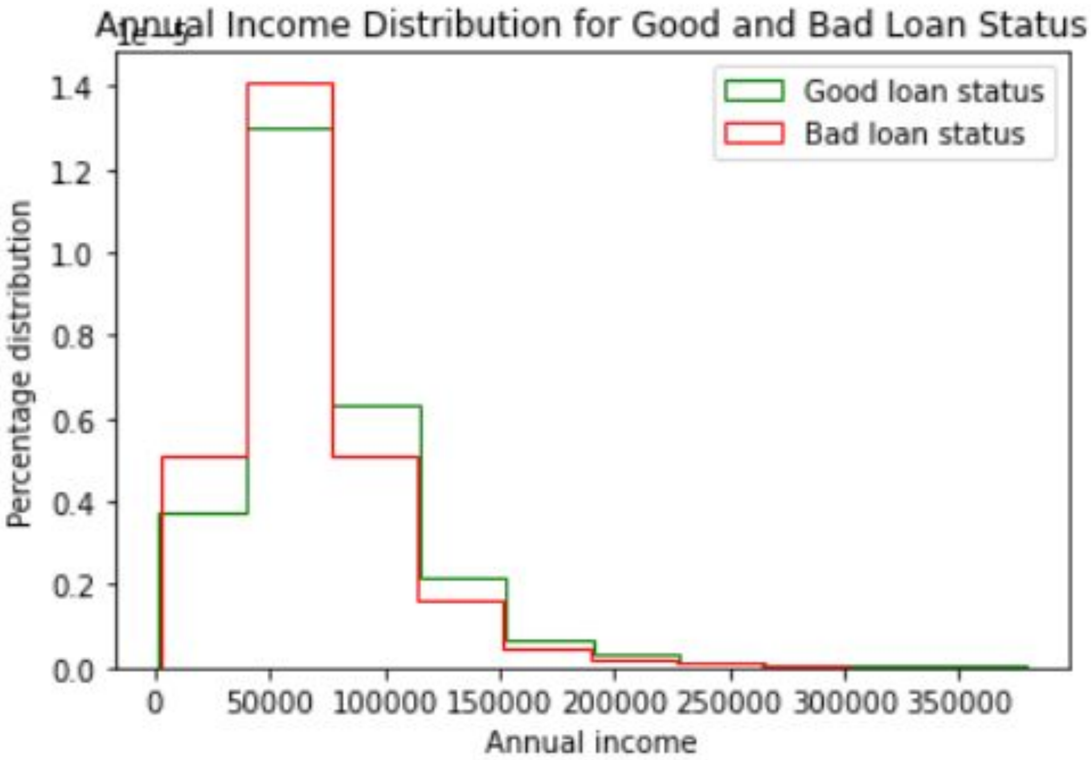
EDA - Interest Rate of Borrowers in Bad Loan Status is Significantly Higher



EDA - Bad Loan Statuses Correspond with Higher Median DTI Value



EDA - Lower DTI Corresponds with Higher Annual Income



EDA - Zip Codes with Highest Default Ratio

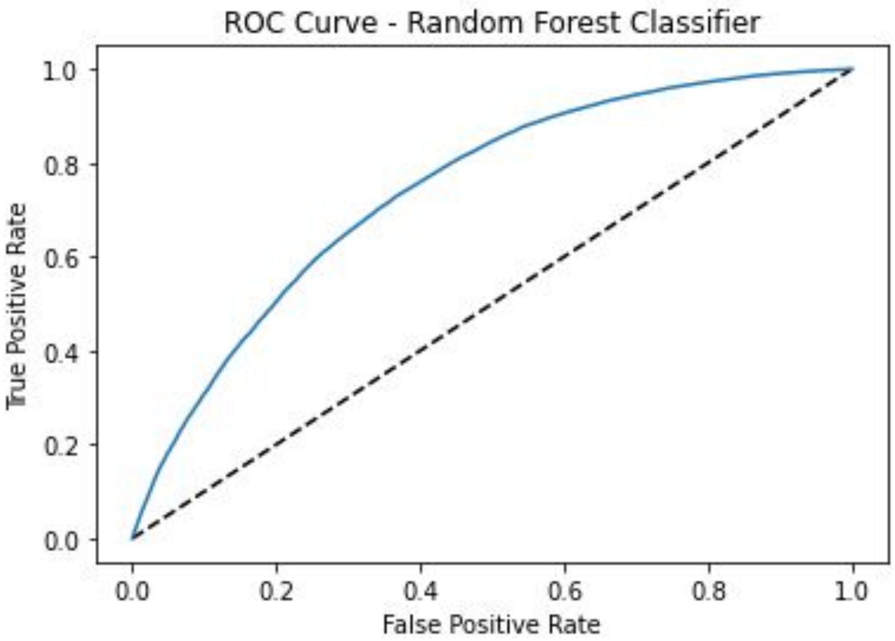
Zip Code	Loans in Bad Status	Loan Count	% in Bad Status
415xx	12	75	0.160
736xx	12	83	0.144
237xx	26	191	0.136
126xx	28	209	0.133
638xx	20	154	0.129
668xx	13	105	0.123

Preprocessing & Training Data Development

- Grade and sub-grade columns columns were encoded as numeric columns.
- The Categorical columns were encoded using One Hot Encoding.
- The numeric columns were standardized using StandardScaler.
- Target and Predictor Variables.
  - Loan status was chosen as the target variable (y).
  - The rest of the columns became the predictor variables (X).
- The data was split into Training Set, X_train, y_train (80%) and Test Set, X_test, y_test (20%).

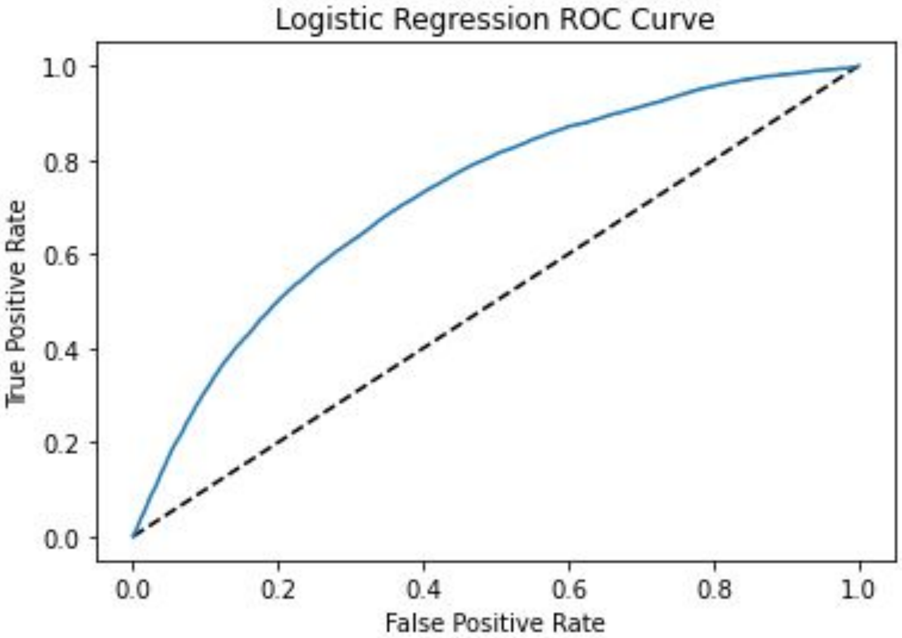
Modeling - Random Forest Classifier

- ➔ Chosen n-estimator value: 500
- ➔ ROC/AUC Score: 0.74



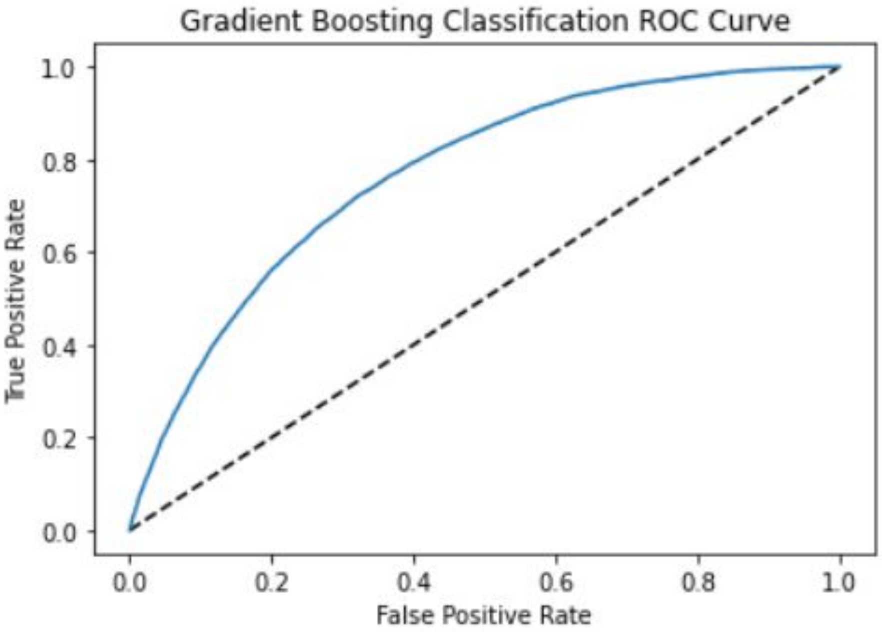
Modeling - Logistic Regression

➔ ROC/AUC Score: 0.72



Modeling - Gradient Boosting

- ➔ n_estimator value: 600
- ➔ Max_depth: 3
- ➔ ROC AUC Score: 0.77



Best Performing Model - Gradient Boosting

Model Comparison Table

	Random Forest	Logistic Regression	Gradient Boosting
ROC/AUC Score	0.74	0.72	0.77
Optimal threshold	0.072	0.065	0.074
F1 score with the optimal threshold value	Class 0 0.78 Class 1 0.22	Class 0 0.76 Class 1 0.21	Class 0 0.80 Class 1 0.24
Confusion matrix with the default 0.5 threshold	[[152450 71] [11557 33]]	[[152520 1] [11590 0]]	[[152487 34] [11572 18]]
Confusion Matrix with the optimal threshold	[[99246 53275] [3498 8092]]	[96218 56303] [3444 8146]	[[103940 48581] [3276 8314]]

Best performing model - Gradient Boosting

➔ Best ROC/AUC score and F1 Score.

➔ Minimizes the false positives while also keeping the false negatives low.

Planned Improvements

- Try PCA dimensionality reduction to see if the performance of the model will improve.
- Try resampling method to handle the imbalance in the data.

“All models are wrong, but some models are useful.”

George E. P. Box

Project Files

Project Notebooks
Project Report

Special Thanks

- For Husain Battiwala for making the data available on Kaggle!
- For Tony Paek for his amazing mentorship!
- For my husband and boys for their encouragement and support!