# Foundations of Data Science

## CS-6053 Fall 2017

## Prof. Rumi Chunara

**TA: Josua Krause**


## Term Project

## Loan Default Prediction

**By Group of**

**Harish Puvvada: hp1047**

**Vamsi Mohan Ramineedi:vmr286**

# Loan Default Prediction

## Business Understanding:

Lending Club is a US peer-to-peer lending company, headquartered in San Francisco, California. It was the first peer-to-peer lender to register its offerings as securities with the Securities and Exchange Commission (SEC), and to offer loan trading on a secondary market. Lending Club operates an online lending platform that enables borrowers to obtain a loan, and investors to purchase notes backed by payments made on loans. Lending Club is the world's largest peer-to-peer lending platform.

Lending Club enables borrowers to create unsecured personal loans between $1,000 and $40,000. The standard loan period is three years. Investors can search and browse the loan listings on Lending Club website and select loans that they want to invest in based on the information supplied about the borrower, amount of loan, loan grade, and loan purpose. Investors make money from interest. Lending Club makes money by charging borrowers an origination fee and investors a service fee.

## Motivation:

People often save their money in the banks which offer security but with lower interest rates. Lending Club operates an online lending platform that enables borrowers to obtain a loan, and investors to purchase notes backed by payments made on loans. It is transforming the banking system to make credit more affordable and investing more rewarding. But this comes with a high risk of borrowers defaulting the loans. Hence there is a need to classify each borrower as defaulter or not using the data collected when the loan has been given.

## Problem Statement:

To **classify** if the borrower will default the loan using borrower's finance history. That means, given a set of new predictor variables, we need to predict the target variable as 1 -> Defaulter or 0 -> Non-Defaulter. The metric we use to choose the best model is 'False Negative Rate'. (predictor and target variables explained later)

## Data Engineering:

Lending Club maintains all its data year-wise. For this project, we have collected data from the lending club website for the years 2012-14. The dataset consists of 360,000 observations and 145 features. Out of the 145 features in our dataset, many of them were empty. We have removed all such features. Also, the features which didn't seem relevant to our goal were removed.

The dataset had many empty and irrelevant features (personal details of borrowers were not disclosed by the company). They have been removed.

- String values have been formatted to integers.
- Categorical values have been transformed to numericals.
- Redundant variables have been dropped.
- Filled NAN values with mean values of corresponding columns.
- All the numerical values have been scaled to a range between -1 and 1.

Now, using correlation matrix of the dataset and in-depth reference to Lending Club's data dictionary, we chose 20 best features related to our objective. Now the dataset shape has been reduced to (140000,20).

## Predictor Variables:

On the above 20 features, we have implemented Recursive Feature Elimination (RFE) using Logistic Regression model to get the best 10 features.

Below mentioned are the features used for our model:

| Predictor Variables | Description |
|---|---|
| funded_amnt | The total amount committed to that loan at that point in time. |
| emp_length | Employment length in years. Possible values are between 0 and 10 where 0 means less than one year and 10 means ten or more years. |
| annual_inc | The self-reported annual income provided by the borrower during registration. |
| last_pymnt_amnt | Last total payment amount received. |
| mort_acc | Number of mortgage accounts. |
| int_rate | Interest Rate on the loan. |
| mo_sin_old_rev_tl_open | Months since oldest revolving account opened. |
| avg_cur_bal | Average current balance of all accounts. |
| acc_open_past_24mths | Number of trades opened in past 24 months. |
| Num_sats | Number of satisfactory accounts. |

## Target Variable:

The target variable in our dataset is '**loan_status**' which shows the status of the loan. It has 3 different values – 'Charged Off', 'Fully Paid' and 'Default'.

**Fully Paid:** Loan has been fully repaid.

**Default**: Loan has not been current for 121 days or more.

**Charged Off**: Loan for which there is no longer a reasonable expectation of further payments.

Since our project goal is to predict whether a borrower will default the loan, we are considering only the observations where loan_status is either Fully Paid or Charged Off. We have changed Fully Paid as 0 and Charged Off as 1 where 1 indicates the borrower as a defaulter.

## Models Applied and Motivation:

### Random Forests Classification:

We have a feature '**last_pymnt_amnt'** which has an importance of more than 30%. Random forests select a subset of features in each of its decision trees thereby reducing the bias (because of high importance of single feature) of the model. The final output will be the mode of the outputs of all its decision trees which has better results than decision trees (which can possibly overfit). Hence, we chose to start our classification with random forests.

### MultiLayer Perceptron:

MLP utilizes backpropagation for training. Its multiple layers and non-linear activation function help us distinguish data that is not linearly separable.

### Support Vector Machine:

We choose to build SVM classifier because, once a hyperplane is found, most of the data other than the support vectors (which are points closest to the boundary) become redundant. This means that small changes to data cannot greatly affect the hyperplane and hence the SVM. So, Support vector machine tends to generalize very well.

### Logistic Regression:

With logistic regression, outputs have a nice probabilistic interpretation, and the algorithm can be regularized to avoid overfitting. Hence, we choose to build logistic regression classifier.

### Hyper Parameter tuning and Advanced Algorithms:

We have implemented all the above models using Randomized and Grid search cross-validation techniques to choose the best hyper-parameters. This showed a rise in accuracy by ~3%.

We implemented,

Bagging – building multiple models (typically of same type) from different sub-samples of training dataset.

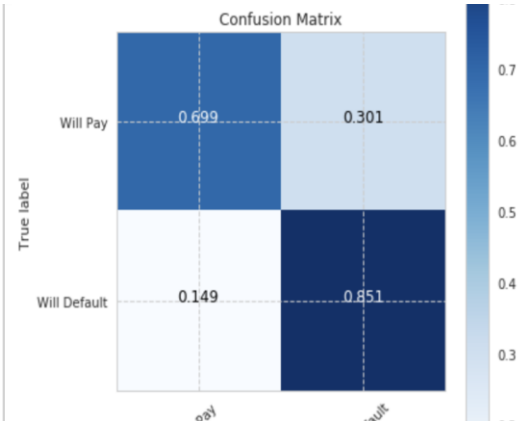
Boosting – building multiple models (typically of same type) each of which learns to fix the prediction errors of a prior model in the chain.

These methods showed negligible rise in the accuracies.

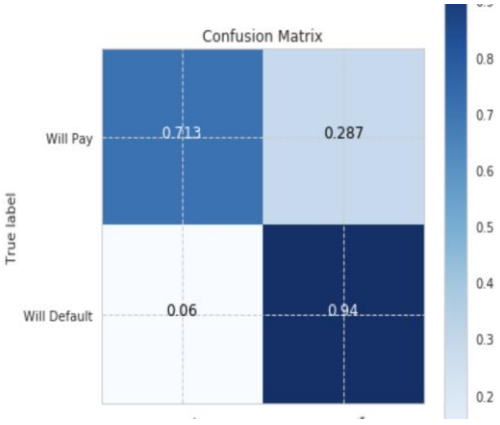
## Evaluation Approach:

For our Loan default prediction project, False Negatives Rate is the best metric to evaluate the model. Lower the number of false negatives, better the model is. In this project, False negative is when model predicting "a borrower will not default a loan even though he will ". Our model cannot afford having higher False Negatives as it leads to negative impact on the investors and the credibility of the company. So, we evaluated our models using the number of False negatives and accuracies.

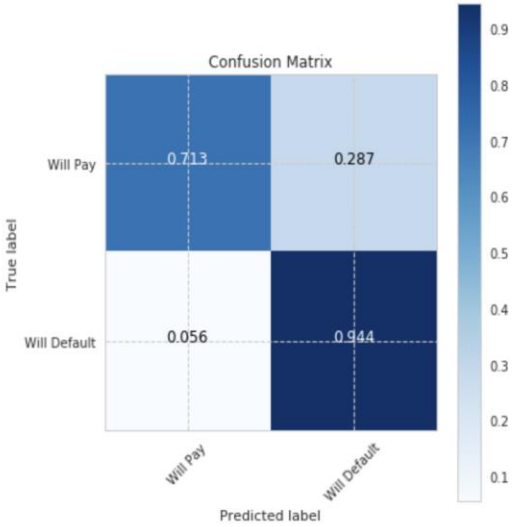
## KNN Confusion Matrix



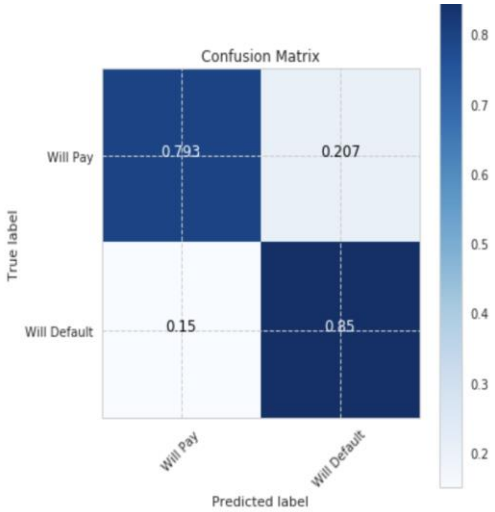

## SVM Confusion Matrix



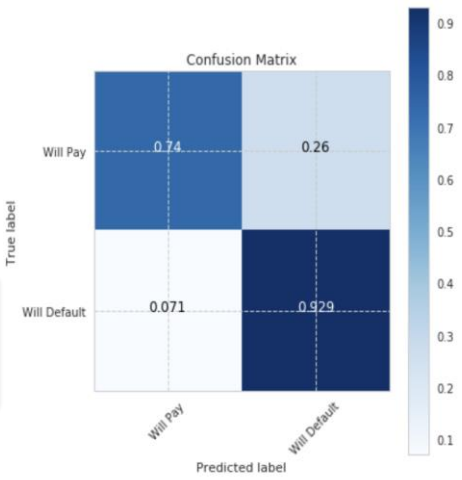

## Logistic Regression Confusion Matrix




## Random Forest Confusion Matrix

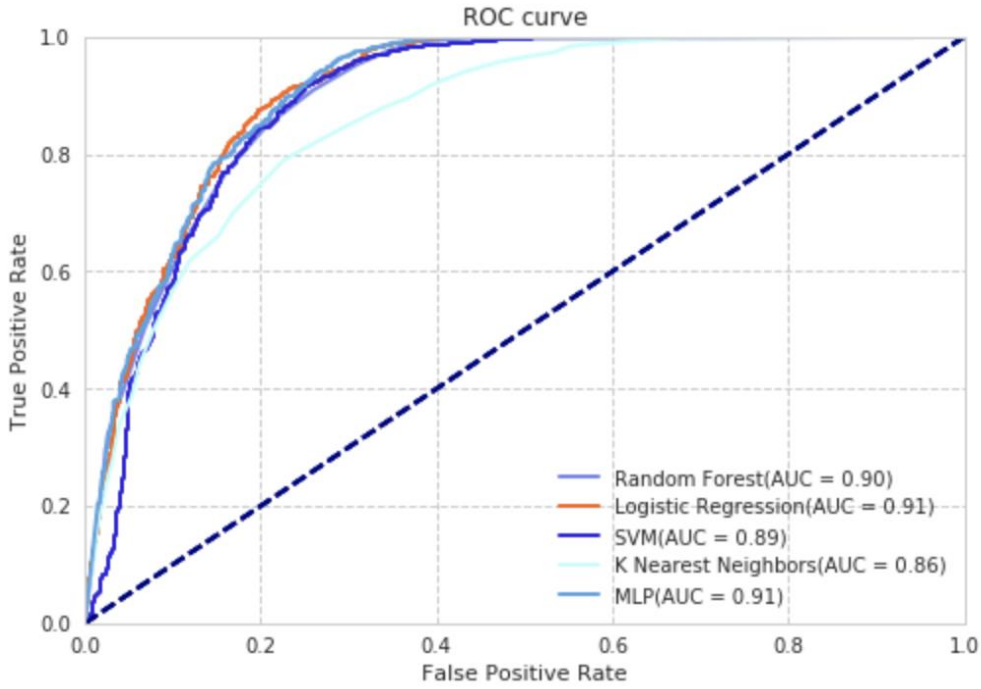



## MultiLayer Perceptron Confusion Matrix




Note: The bottom left cell represents False Negative Rate.

Receiver Operating Characteristic Curve:




**Logistic Regression and MLP models have highest roc_auc_score**

## Model Accuracies:

| | | |
|---|---|---|
| Random Forest with Randomized search CV | ----- | 82.09 |
| Logistic Regression with Grid search CV | ----- | 83.18 |
| Support Vector Machine with Grid search CV | ----- | 82.50 |
| K Nearest Neighbors with Grid search CV | ----- | 77.40 |
| Bagging with Base estimator as Random Forest | ----- | 84.10 |
| Bagging with Base estimator as Logistic Regression | ----- | 83.10 |
| AdaBoost Classifier | ----- | 83.60 |
| MultilLayer Perceptron Classifier | ----- | 83.40 |

Although all the algorithms except KNN have almost same accuracy, their False Negative rates differ which is our main evaluation metric. From the confusion matrices, we can infer that **Logistic Regression model had the least False Negative rate (FNR).**

## Assumptions:

- We have not made any direct assumptions about our data or models as per our best knowledge.

## Limitations:

- We have not considered the effect of delayed payments which can be derived from various payment_dates fields of the dataset.
- Cross-validation for support vector machine modeled here requires high computational time.

## Problem in Scope of Class:

- Our problem can be expressed as Supervised learning problem where we have target variable specified. We have used supervised learning algorithms like K-nearest neighbors, support vector machines and logistic regression which were discussed in the class.
- We had 140,000 observations of data, which took a lot of time for training. By implementing learning curve for our data, we realized that our models do not learn after 7000 observations. So, we downsized it. Thanks to the lecture on Feature Selection and Data Preparation.
- Our data has been preprocessed using techniques like scaling, filling null values which were discussed in the class and also in the assignments.

## Change from Original Proposal:

From our dataset, we had the option of building 2 types of predictive models with target variables being,

1. if a person can be granted a loan or not
2. if a person who already borrowed the loan, will repay it or not.

Initially, we planned to build a predictive model of type 1 (as defined above). After data exploration, building a model of type 2 seemed more appropriate as the data had more relevant features which can predict well about repaying. The model of type 1 is giving accuracies of less than 50% which is highly unreliable. (Note: We have used type 2 for the mid-presentation)

## Points Division between Team Members

We both have taken keen interest in the project and have contributed equally. So, we distribute the 10 points equally as 5 for each of us.

Harish Puvvada: hp1047 – 5 points

Vamsi Mohan Ramineedi: vmr286 – 5 points

# Citations and Bibliography


- http://scikit-learn.org/stable/.
- Few tutorials on cross validation from YouTube channel Data School https://www.youtube.com/user/dataschool .
- http://seaborn.pydata.org/ .
- Advanced algorithms like Boosting and Bagging from Machine Learning Mastery https://machinelearningmastery.com/ .