# Exploratory Analysis and Prediction - Lending Club Loan Data

**MOHAMMED SHOEBUDDIN HABEEB**
mhabeeb2@student.gsu.edu

**DIVYA BHUVANAPALLI**
dbhuvanapalli1@student.gsu.edu

## Abstract

Machine learning has been having a significant impact on financial lending in the recent times. It as turned the tide on traditional lending, allowing for more accurate and faster decisions. From analysis of individuals to analysis of trends and patterns of the customers machine learning is extensively being used. In this project, we plan to use the data from Lending Club, a well-known peer-to-peer lending platform based in San Francisco, California, to build statistical models that use various attributes of the borrowers to predict the grades given to loan applications by the Lending Club. A loan is assigned a grade ranging from A through F by taking into account various attributes like credit score, several indicators of credit risk from the credit report and loan application. Based on the grade assigned to the loan, the term of repayment, interest rates etc are determined.

Keywords— Data Mining, Loan classification, Machine Learning

## 1 Introduction

Machine Learning and Data Mining - along with Artificial Intelligence - were once the promising future of Computer Science. It is safe to say that today it looks more than just promising, and a little bit closer to the present than to the future. Thanks to the advancement in storing technologies, and computational power, what was once theoretical foundations for extracting relevant insights from Data; and being able to predict new information based on historical data, is what's driving today's business decisions, what is the hope for medical advancements, understanding complex data such as that from space, image recognition, and many other interesting and relevant applications use Data Mining/Machine Learning algorithms as their core and driving force.

In finance, a loan is the lending of money by one or more individuals, organizations, or other entities to other individuals, organizations, etc. The recipient (i.e. the borrower) incurs a debt, and is usually liable to pay interest on that debt until it is repaid, and also to repay the principal amount borrowed. The whole process of ascertaining if a borrower would pay back loans might be tedious hence the need to automate the procedure. The two most critical questions in the lending industry are how risky is the borrower and given the borrower's risk, should we lend money. Lending club is the largest peer-to-peer marketplace connecting borrowers with lenders. Borrowers apply through an online platform where they are assigned an internal score. Lenders decide 1) whether to lend and 2) the terms of loan, such as interest rate, monthly instalment, tenure etc. Some popular products are credit card loans, debt consolidation loans, house loans, car loans etc. There are many features that contain a widely varied and can be classified into three categories: 1) Customer's demographics : such as employment length, title, annual income, zip code, etc. 2) Loan information : such as loan amount, funded amount, interest rate, loan status, loan grade, etc. 3) Characteristics, and customer behavior variables : Credit history information, loan purpose, application type.

Like most other lending companies, lending loans to 'risky' applicants is the largest source of financial loss. This credit loss is the amount of money lost by the lender when the borrower refuses to pay or runs away with the money owed. In other words, borrowers who default cause the largest amount of loss to the lenders. In this case, the customers labelled as 'charged-off' are the 'defaulters'. If one is able to identify these risky loan applicants, then such loans can be reduced thereby cutting down the amount of credit loss.

## 2 Methodology

### CRISP-DM : Cross Industry Standard Process Data Mining

In this project we are going to try to go through all the steps that involve a machine learning/data mining workflow right from the start, when we make the decision of which data set to use, all the way to the conclusions and further recommendations after we have evaluated the data. First let's begin by trying to define the concepts. Data Mining and Machine Learning are terms which some might even use interchangeably to describe processes and activities that are . One way of looking at it is that both are meant to obtain insights from data, the difference is that the Data Mining will be done by the user actively applying functions and algorithms to obtain insights, while Machine Learning the user allows the model to do the predictions or analysis by itself. In any case, the workflow can be outlined in different ways, but the high level idea is mostly about the following: 1) Business understanding 2) Data understanding 3) Data preparation 4) Modeling 5) Evaluation 6) Deployment.

Out of these one would think that the most important and the ones that one should spend more time on are the modeling, or evaluation parts. But as it turns out, the more lengthy - and sometimes tedious - part is Data Preparation, but even this can't be done unless there is accurate business understanding and the subsequent data understanding. There is an expression " Garbage In, Garbage Out ". This means that if the quality of the input is poor, then the results and insights derived out of it are more probably equally poor. Is for this that the most important steps are the Business and Data understanding, followed by the data preparation. The first thing we needed to do was to decide which data set to use, this can be tricky because of one of the most important things that are needed to be able to derive relevant insights from data, that is " Domain knowledge ". It is the understanding, skills, insightfulness, and information on a particular subject, which is crucial to asking the right questions, and interpreting the data at hand.

## 3 Exploratory Data Analysis

### 3.1 Data Exploration

The original dataset is available for download on the Lending Club website. So far the data is available till 2020 Q1 and can be dated back to 2007. For our analysis, we consider data for the years 2007-2011. The dataset contains information on almost all the loans issued by LC, except a few that LC was not authorized to release publicly. From initial look into the data we see that more than 42000 records of 147 attributes associated with the lending club. From the point of view of the customer, the grade they might receive will affect their interest rate on the loan, so let's try to predict the grade of the customers based on their data and see if that can be done. From the data on the grades we can see that there are seven different grades that can be given to a customer - A through F with A being the highest as shown in 3 this presents a problem because models tend to give better results the less number of targets are in the mix, what's called "cardinality". So these were encoded into three separate categories(High,Medium,Low) as shown in 4 to help improve this. It should be pointed out that there are some sub grades as well, but this can be worked in a similar manner if needed.

The next step is to continue to clean and prepare the

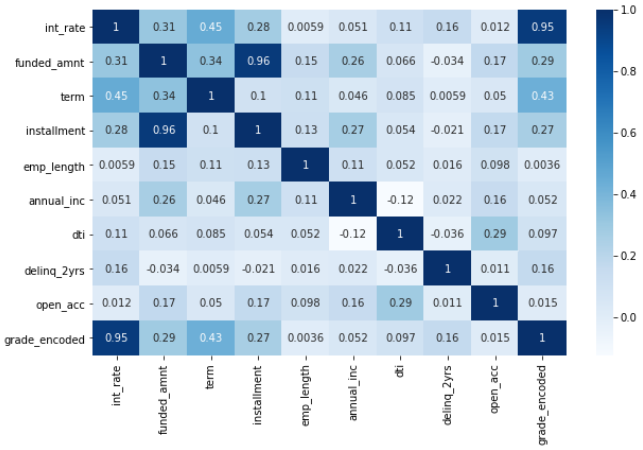


Figure 1: Correlation Plot

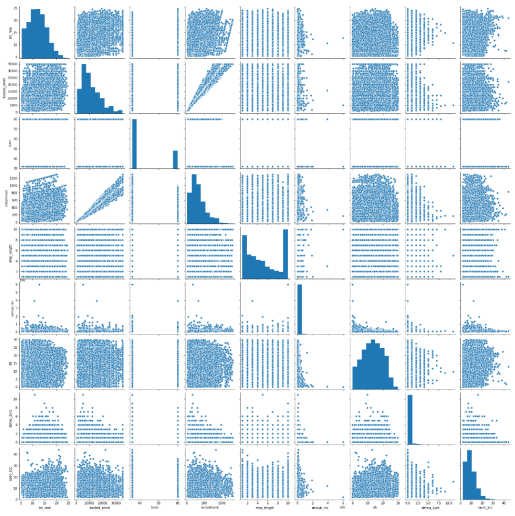

Figure 2: Pairwise Plot

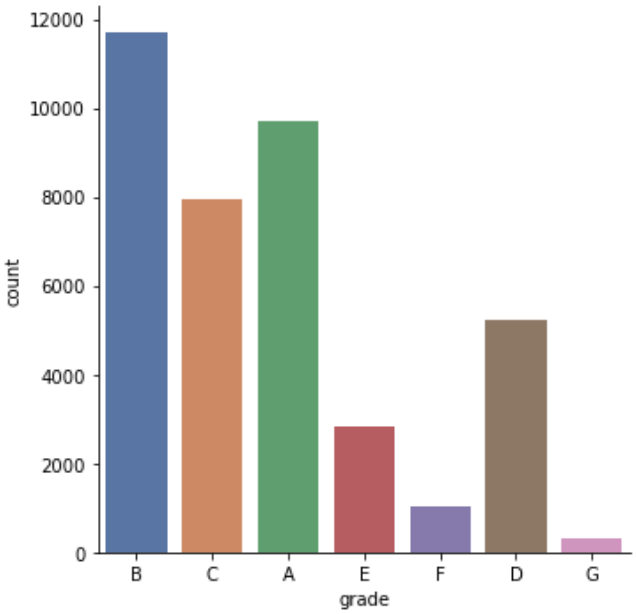


Figure 3: Distribution of Grades

data. Three major things were done: Removing the null values; binning numerical into categorical data, and one hot encoding which was done for the grades

#### 3.1.1 Removing NULL values

Many of these attribute columns had overwhelming number of NULL values. Initially we remove all the columns which have complete NULL values as they do not add anything to our Analysis.

The null values are points with missing records for a given feature on a given row. These cannot be added as input to the model and this becomes a problem which has to be carefully handled. Ideally by removing the null values, we will still have enough information to get a representative sample of the original data. First we dropped the columns which had only null values, which also reduces the dimensionality of the data from 147 features to 65 features, which as we will explain later actually helps decrease the computational cost of running the more computationally exhaustive models such as KNN.

This still leaves columns with a great amount of null values, so columns with more than 60% of null values were removed. This threshold is somewhat arbitrary, but provides a reasonable cutting point which can help reduce the number of null values, this takes the number of features down to 56. After this there are still some columns with null values, one option could be to impute values such as fill with zeros, or even the mean, median or other central tendency values. Another could be to remove only those rows that have null

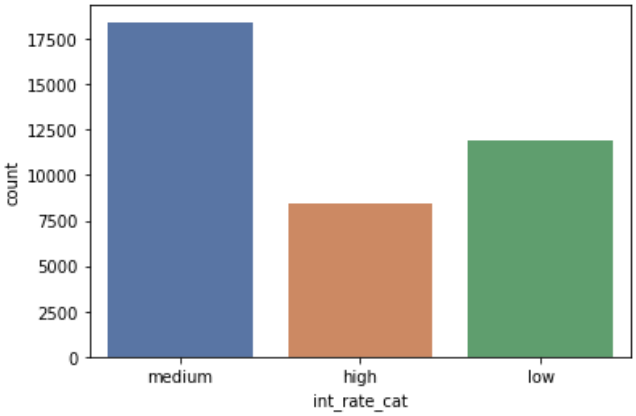


Figure 4: Distribution of Modified Grades

values, but this might dramatically reduce the number of records present in the data. The alternative that we used was to analyze the features and decide which ones should be the ones to be used, leaving the ones that are typically more directly affecting the possible grade that a borrower obtains. This serves the purpose of making the model less computationally demanding, as some of the less greedy algorithms will take more time to compute as the number of features increase. This is sometimes called "The curse of dimensionality". A more realistic approach in which computational power is not a constraint can certainly include all the features, as long as they are properly pre-processed, and this might provide better results overall.

#### 3.1.2 Converting Numerical to Categorical

Once the features were selected, those with categorical data were transformed into numerical. Again this is because the models are not typically built to receive "strings" as input, but rather numerical data. Converting all the columns to float gives a sense of uniformity to the model and avoids any mismatch error building up.

#### 3.1.3 One hot encoding

When presented with categorical data that is not binary, all sorts of problems occur when running the model, one way of addressing this is to map them as binary by means of a process called "One-hot encoding", which consists in visualizing the different labels as either you are label x or not, instead of labelling it as t,u,v,x,y,z. This will allow for the models to be more robust.

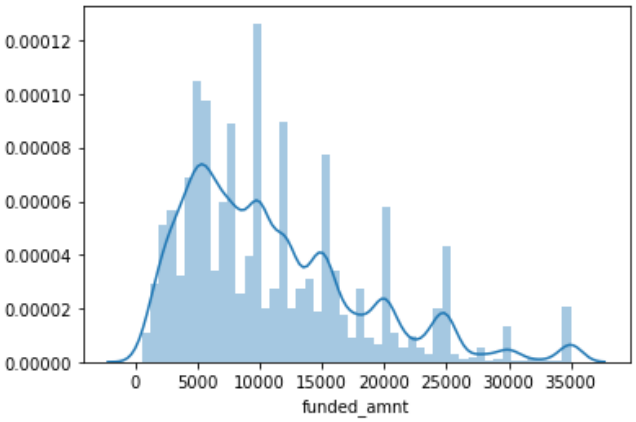


Figure 5: Distribution of Amount

#### 3.1.4 Understanding the data

Once we get through all these preprocessing steps we remove some columns which are not relevant or are identified as ones not adding any value to the model training. Columns like Zip Code, Sub Grade, Address state and other s are removed to make a final dataset of 35 feature attributes and more than 38000 records.

One crucial step to understand the data is to perform univariate and bivariate analysis by means of visualizations. The pair plot and the correlation matrix 1 shows that there are very few variables that have a linear relation, only the installment vs funded amount, which of course makes sense since the installment will be proportional to the amount that was borrowed. Then the interest rate with the given grade, which again makes sense because there will be a strong correlation as lenders typically will give lower interest rates to those who are better qualified.

To understand our data we see distribution of how much loan amounts are typically given and we see from 5 that most of the loans are in the range of 4000 to 15000. These loans classified as grades are modified for the ease of computation to 3 classes as mentioned earlier where initial alphabetic grades A and B are considered as 0, C and D are considered 1 and the rest are merged to 2 after encoding. 4 shows us the count of all the modified grade classes where medium grades dominate the distribution. Another peculiar we see in the pairplots in 2 is the distribution of term columns which is predefined as either a 36 month loan or a 60 month loan. This is also encoded to make sure the number are representative instead of ordinal.

## 4 Modeling

### 4.1 Supervised Machine Learning

Supervised learning is the most common subbranch of machine learning today. Supervised machine learning algorithms are designed to learn by example. When training a supervised learning algorithm, the training data will consist of inputs paired with the correct outputs. During training, the algorithm will search for patterns in the data that correlate with the desired outputs. After training, a supervised learning algorithm will take in new unseen inputs and will determine which label the new inputs will be classified as based on prior training data. The objective of a supervised learning model is to predict the correct label for newly presented input data.

Supervised learning can be split into two subcategories: Classification and regression.

#### 4.1.1 Classification

During training, a classification algorithm will be given data points with an assigned category. The job of a classification algorithm is to then take an input value and assign it a class, or category, that it fits into based on the training data provided.

Classification problems can be solved with a numerous amount of algorithms. Whichever algorithm you choose to use depends on the data and the situation. Here are a few classification algorithms that we used to classify the grades in our loan data:

Random Forest: Random forest, like its name implies, consists of a large number of individual decision trees that operate as an ensemble. Each individual tree in the random forest spits out a class prediction and the class with the most votes becomes our model's prediction.

Gaussian Naive Bayes: A Naive Bayes classifier is a probabilistic machine learning model that's used for classification task. The classifier is based on the Bayes theorem.

Bayes Theorem: $P(A|B) = \frac{P(B|A)P(A)}{P(B)}$

Bayes theorem estimates the probability of an event, based on prior knowledge of conditions that might be related to the event. We can find the probability of A happening, given that B has occurred. Here, B is the evidence and A is the hypothesis. The assumption made here is that the features are independent implying the presence of one particular feature does not affect the other. Hence it is called naive.

| Model | Accuracy | Hyperparameters |
|---|---|---|
| $LogisticRegression$ | 0.611 | Kernel : lbfgs |
| $KNearestNeighbors$ | 0.588 | K : 11 |
| $SupportVectorMachines$ | 0.602 | Gamma : Scale |
| $RandomForest$ | 0.620 | Estimators : 1000 |
| $AdaBoost(RF)$ | 0.623 | Estimators : 1000 |
| $PrincipleComponentAnalysis$ | 0.538 | 12 Component LR |
| $NeuralNetwork$ | 0.597 | 4 layered |

Table 1: Model Performance Comparison.

Support Vector Machines: A support vector machine (SVM) is a supervised machine learning model that uses classification algorithms for two-group classification problems. For multiclass classification problem like ours, the same principle is utilized. The problem is broken down to multiple binary classification cases as one-vs-one or one-vs-rest.

The one-vs-rest method basically divides the data points in class x and rest. Consecutively a certain class is distinguished from all other classes.

The number of classifiers necessary for one-vs-one multiclass classification can be retrieved with the following formula (with n being the number of classes): n*(n-1)/2

In the one-vs-one approach, each classifier separates points of two different classes and comprising all one-vs-one classifiers leads to a multiclass classifier.

We use the one-vs-rest approach in our classification.

K-Nearest Neighbor: The KNN algorithm assumes that similar things exist in close proximity. In other words, similar things are near to each other. An initial value for K is assumed and the data is grouped into k-groups. The distance between the new data point and the rest is calculated. This is repeated several times to get the best possible results.

AdaBoost Classifier: AdaBoost is similar to Random Forest classifier in that they both tally up the predictions made by each decision trees within the forest to decide on the final classification. In AdaBoost, the decision trees have a depth of 1 (i.e. 2 leaves). In addition, the predictions made by each decision tree have varying impact on the final prediction made by the model.

#### 4.1.2 Regression:

Regression is a predictive statistical process where the model attempts to find the important relationship between dependent and independent variables. The goal of a regression algorithm is to predict a continuous number such as sales, income, and test scores.

There are many different types of regression algorithms. The most common regression methods used for classifications problems is Logistic Regression.

Logistic Regression: Logistic regression is another technique borrowed by machine learning from the field of statistics. It is the go-to method for binary classification problems (problems with two class values).For a problem like grade classification with multiple classes multinomial logistic regression can be used. It is a form of logistic regression used to predict a target variable have more than 2 classes. It is a modification of logistic regression using the softmax function instead of the sigmoid function the cross entropy loss function. The softmax function squashes all values to the range [0,1] and the sum of the elements is 1.

### 4.2 Unsupervised Machine Learning

Unsupervised Learning is a machine learning technique in which the users do not need to supervise the model. Instead, it allows the model to work on its own to discover patterns and information that was previously undetected.

Here are a few algorithms that we used to classify the grades in our loan data:

**Principal Component Analysis:**

Principal Component Analysis (PCA) is an unsupervised, non-parametric statistical technique primarily used for dimensionality reduction in machine learning. High dimensionality means that the dataset has a large number of features. The primary problem associated with high-dimensionality in the machine learning field is model overfitting, which reduces the ability to generalize beyond the examples in the training set.

From Figure 6, we can conclude that only 12 compo-

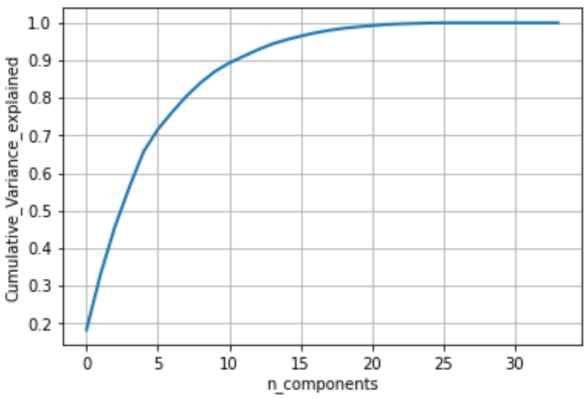


Figure 6: PCA Component Variance

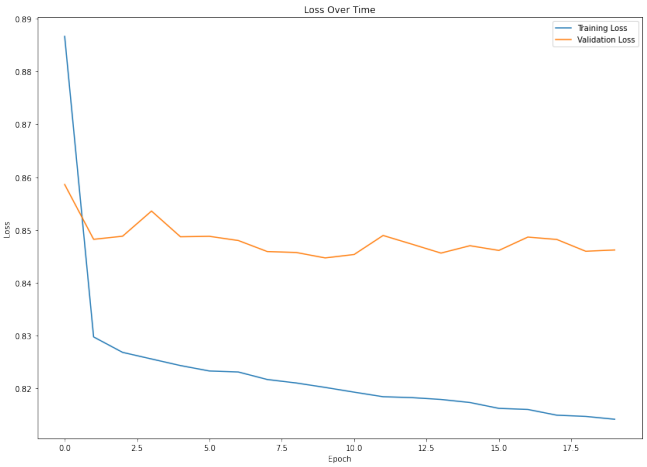


Figure 7: Training and Test Loss

nents are required to get 90 percent of the variance.

### 4.3 Neural Network:

Neural networks are widely used in Machine learning to learn better representations of the input data. We use a simple network with four layers with 64 neurons each activated by Rectified Linear Unit(ReLU) in first 2 layers. It is also followed by a softmax layer that converts the output to class probabilities. With an in input of 12 components, batchsize of 32 and 20 epochs. Major advantage of using ReLU activation function over Tanh function is to avoid diminishing gradient and keep gradient centered. Adam Optimizer is used instead of SGD to update weights in every iteration because it effectively combats the issue of saddle points.

### 4.4 Results

We see standard accuracy performances on all the classifiers as the data is very diverse and has class imbalance issues. Naive Bayes classifier returned with an accuracy of 33% which was majorly because of the naive assumption of independence amongst the columns. We disregard Naive Bayes to consider KNN which at first had a similat performance to NB but when iterated through a GridSearch hyperparameter tuning returned a k value of 11 and Euclidean distance. This dramatically increased the accuracy of the KNN model. Random Forest classifer gives a similar result as KNN but when an AdaBoost method is applied over Random Forest the performance scores are bettered. Logistic Regression does not necessarily improve on the score so we went for applying PCA analysis of the data to see the variance distribution. We see from Figure 6 that 90% of our data can be explained by 12 components so we just train our initial Logistic Regression model but that does not improve the score. In a final attempt to make sense of the data we built a 4 layered Neural Network on Tensorflow and Keras frameworks. This 4 layered model is trained over 20 epochs and we see the training and validation errors in Figure 7. The stagnating validation error suggests that the model is overfitting on training data and underfitting on Validation error. The following table, Table 1 illustrates the accuracy's obtained using different hyper parameters for all the classifiers used in this project allowing comparison of the results. We notice that Random FOrest classifier with an Adaboost is the best performing model.

## 5 Conclusion

We see that all the ML classifiers have average performances as the data set is real world problem. Several techniques were tried like GridSearchCV on some classifiers to identify and tune the hyper parameters to improve the performance. By reducing the components (less than 5), we were able to improve the accuracy but this also induces high bias in the data. An alternate to improve the model performance would be to use larger data sets. Neural network is tuned with training but this could be further improved by increasing the number of layers. Deeper Neural networks although successful exponentially increase the parameters and the feature space in turn drastically increasing the time and cost associated with the dataset. We think that class imbalance can be changed by implementing stratified undersampling to manage classes. We can also use the original class labels A to F instead of merging them to get a more distributed classes but at the expense of increase model complexity. Moving away

from model fitting we can also use alternate evaluation techniques like F1 Score or ROC to get a better understanding of model performances. This project was an insightful introduction to the application of Data Mining in real world scenario which helped us get comfortable with applying complex Machine Learning and Data Mining techniques along with identifying potential work that can be done to improve the performances.

## References

[1] https://towardsdatascience.com/crisp-dm-methodology-leader-in-data-mining-and-big-data-467efd3d3781

[2] https://towardsdatascience.com/random-forest-classification-and-its-implementation-d5d840dbead0: :text=Random

[3] J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[4] I. S. Jacobs and C. P. Bean, "Fine particles, thin films and exchange anisotropy," in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.

[5] https://www.guru99.com/unsupervised-machine-learning.html

[6] https://towardsdatascience.com/multiclass-classification-with-support-vector-machines-svm-kernel-trick-kernel-functions-f9d5377d6f02

[7] https://towardsdatascience.com/machine-learning-part-17-boosting-algorithms-adaboost-in-python-d00faac6c464

[8] https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761