# Building A Credit Scoring Model To Assign A Reference Score Based On Credit Transaction And Relevant Profile Data

by

Saqib Al Islam
16101084
Rifah Sama Aziz
19141019
Aritra Ahmed
16101216
Fauzia Abida
16101320

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science and Engineering

Department of Computer Science and Engineering
Brac University
September 2019

# Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

**Student's Full Name & Signature:**

| | |
|---|---|
| Saqib Al Islam<br>16101084 | Aritra Ahmed<br>16101216 |
| Rifah Sama Aziz<br>19141019 | Fauzia Abida<br>16101320 |

# Approval

The thesis/project titled "Building a Credit Scoring Model To Assign a Reference Score Based on Credit Transaction and Relevant Profile Data" submitted by

1. Saqib Al Islam (16101084)
2. Rifah Sama Aziz (19141019)
3. Aritra Ahmed (16101216)
4. Fauzia Abida (16101320)

Of Summer, 2019 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science and Engineering on September 5, 2019.

**Examining Committee:**

Supervisor:
(Member)

Dr. Mahbub Alam Majumdar
Professor
Department of Computer Science and Engineering
BRAC University

Co Supervisor:
(Member)

Md. Saiful Islam
Lecturer
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Dr. Mahbub Alam Majumdar
Professor
Department of Computer Science and Engineering
Brac University

# Ethics Statement

Assigning a score to an individual based on a heuristic provided by a Machine Learning model without any further analysis or evaluation might provoke ethical dilemmas. Hence, we assured the transparency of the evaluation process. In addition to that provide visual interpretation of the output provided by the Machine Learning Model.

# Abstract

A credit score is a numerical expression based on a level analysis of a person's credit files, to represent the creditworthiness of an individual. The credit score plays a major role in banks, financial institutions loaning money to individuals for their personal or business needs. This score is given based on factors such as personal information, assets, financial behavior and financial history. This system is not digitized or implemented yet in Bangladesh. So our aim is to build a reliable and robust credit scoring model which would help institutions like such to have an accurate reference score to rely on when validating a client. We were able to obtain an optimized model with an accuracy of( 93%). The model is based on CART(Classification and Regression Trees) using Gradient Boosting method(GBM). We also proposed a new hybrid model consisting of a two step architecture. The first one based on distributed Random Forests, the individual decision tree outputs of which was fed into a Deep Neural Network(DNN), and trained on to achieve marginally better results than using only Random Forest approach. Since, credit scoring an individual is a sensitive issue, it is not ethical to provide a score without proper justification. We conducted interpret-ability analysis on our model and generated visual representations of the criterion affecting the output of our model and provide necessary information to analyze the client effectively. Our results were conclusive and imitated the process of evaluating an individual precisely. The work-flow we proposed could be implemented in production to provide a concrete base for evaluation and prediction of defaulters. Simultaneously provide a detailed overview of the results obtained. This could help financial institutions immensely and help them save millions lost by default loans.

**Keywords:** Credit Score, Credit Risk, Loan Assessment, Machine Learning, Artificial Intelligence, Random Forests, Gradient Boosting, GBM, Extreme Gradient Boosting, KNN, RF, Deep Neural Networks, DNN, fDNN, Interpret-ability, LIME.

# Dedication

We would like to dedicate this thesis to our loving parents. As well as, all the amazing faculties we encountered and learnt from in the course of pursuing our Bachelors degree. It has been a journey worthwhile...

# Acknowledgement

We would like to acknowledge that this report has been done under the supervision of Dr. Mahbub Alam Majumdar. Firstly, we would like to thank Almighty Allah for providing us the opportunity, guidance and confidence to work with Dr. Mahbub Alam Majumdar on this thesis paper. Secondly, we express sincere gratitude to our supervisor, Dr. Mahbub Alam Majumdar, for assisting and guiding us patiently throughout the year. His valuable suggestions, ideas, support and effort have significantly helped us in our research work. Thirdly, we are thankful to our co-supervisor, Mr. Md. Saiful Islam, who encouraged and assisted us with our work in researching. Lastly, we would like to express genuine appreciation to our parents for their endless support and prayers.

# Table of Contents

# List of Figures

# List of Tables

# Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

$AWS$ Amazon Web Services

$corr$ correlation

$CV$ Cross-Validation

$DNN$ Deep Neural Network

$fi$ Feature Importance

$GBR$ Gradient Boosted Regression

$KNN$ K- Nearest Neighbours

$MAE$ Mean Absolute Error

$MSE$ Mean Squared Error

$P$ Probability

$RF$ Random Forest

$RMSE$ Root Mean Squared Error

$SD$ Standard Deviation

$SDR$ Standard Deviation Reduction

# Chapter 1

# Introduction

## 1.1 Overview

A credit score is a numerical expression which is obtained by analyzing a person's credit files that defines the creditworthiness of an individual. On the other hand, credit scoring is a statistical method to predict the probability of whether a loan applicant or an existing borrower can repay the loan successfully or not.
This method was introduced in the 1950s[49].At present, it is broadly used for consumer lending, credit cards, mortgage lending and so on. Information of the borrowers is collected from their loan applications and from historical credit bureaus. These data include the applicant's monthly income, outstanding debt, financial assets, duration of the same job, whether the applicant has defaulted or failed to pay the previous loan, whether the applicant owns or rents a home, and the type of bank account the applicant has are all potential factors that influence loan performance. Regression analysis relating loan performance to these variables is used to pick out which combination of factors best predicts delinquency or default, and how much weight should be given to each of the factors. Considering the correlations between the factors, it is seen that some of the factors the model developer begins with does not affect much, since they have little value added compared to the other variables in the model. For this reason, according to Fair, Isaac and Company, Inc., a leading developer of scoring models, at the beginning 50 or 60 variables might be considered, but 8 to 12 might end up in the final scorecard as yielding the most predictive combination[49] Again, Anthony Saunders reports that First Data Resources uses 48 factors to estimate the probability of credit card defaults.[4] In most cases, a higher credit score indicates lower risk, and a lender sets a cutoff score based on the amount of risk it is willing to allow.
Some of the statistical methods used to develop credit scoring systems are linear probability models, logit models, probit models, and discriminant analysis models. Two newer methods used to estimate default probabilities include optionspricing theory models and neural networks. Neural networks are artificial intelligence algorithms that allow learning through experience to detect the relationship between borrower characteristics and the probability of default and to determine which characteristics are most important in predicting default. No assumptions have to be made about the functional form of the relationship between characteristics and default probability or about the distributions of the variables or errors of the model. For this reason, this method is more flexible and better than standard statistical

credit scoring methods. Again, an alternative to back-propagation has been used in classification is the probabilistic neural network (PNN) , which involves one-pass learning and can be implemented directly in neural network architecture.[3]
At present, credit scoring is used by many big and small banks for loans under $100,000 in most cases.[4] No scoring model is utilized universally. It has taken longer for scoring to be adopted for business loans, since these loans are less homogeneous than credit card loans and other types of consumer loans and also because the number of this type of lending is smaller, so there is not sufficient amount of data available to make a model. Other organizations, such as mobile phone companies, insurance companies and landlords are also using credit scoring. Digital finance companies such as online lenders also use alternative data sources to calculate the creditworthiness of borrowers.
For this paper our field of interest is assessment of credit risk of loan borrowers. The number of loan defaulters/ charged off loans have been increasing significantly. Transactions are intervened, assets frozen causing huge loss to financial institutions, banks. Reports show that in 2018 alone, China had 9.2 million loan defaulters [39]. The amount of default loans in Bangladesh has increased nearly 3 times since 2011 [42]. In the United States around a million student loans get defaulted [45]. In India, during the period 2013 to 2017 money owed by defaulters quadrupled [35]. Experts also suggest that the current scenario in Bangladesh will hamper the growth of businesses, limit implementation of various strategies to improve employment for the general population [30]. It can be inferred from this that default loans are a burden to the country's economy in addition to having a negative impact on financial institutions. A reliable solution to this issue would be to filter out applications which have a higher risk of defaulting. This can be achieved through a pattern recognition approach, a field of study in which machine learning excels at. Having the ability to recognize underlying pattern present in a certain domain and train on it to improve iteratively. Several studies demonstrated the effectiveness of applying machine learning techniques for credit risk assessment. In [14] the authors have used neural networks and genetic algorithm to prepare a model for credit risk assessment. Besides genetic algorithm, they have tested various other feature selection methods such as forward selection, information gain, gain ratio and Gini index and have concluded that for their data set a combination of neural network and genetic algorithm was the most optimum solution. For a reliable result they have also applied k folds cross validation instead of the train test split. Many other papers[8], [34], [9], [26] conducted supervised learning on credit risk datasets using ensemble methods, tree based models, neural network models, conducting comparative analysis between them and presenting promising results in predicting credit risk
This paper will discuss the application of different supervised algorithms along with feature selection methods to predict a representative score based on an individual profile. Our data set includes personal history along with credit history of an applicant. Regressors such as Gradient boosted regression, extreme gradient boosting, random forest, linear regression as well as a proposed hybrid model consisting of Random Forest Regressor and Deep Neural Network will be used to identify the underlying patterns that exist in the previous borrowers and their credit score, supervised learning would be carried out to generate scores for unseen data. Feature Selection using correlations and feature engineering from instantiating log and sq

rt of numeric columns would be conducted to find the optimum features . Later on 4 folds cross validation randomized and grid search will be used to select the optimum hyper-parameters for selected models. A comparative study of each model will be made to select the most optimum model for credit risk assessment. Model Interpretation would be conducted using LIME [36] during inference time. This will provide insight into the working of the model and provide substantial information for human evaluation if necessary.
The rest of the paper includes brief discussion about some relevant work that has been done in this field. This will be followed by a detailed description of our proposed model and our data set. The later sections will discuss the steps of data pre-processing, the results and experimental analysis and finally the paper will be concluded with future works and concluding remarks.

## 1.2 Importance/Usefulness of credit scoring

Credit score of an individual is characterized from the underlying history of his financial life. Credit scoring incorporates a huge role in financial situations as it is not only restricted to loan approval or credit cards. This score is defined as a statistical method that determines the likelihood of an individual to pay back any money that has been borrowed. The factors of this score vary geographically and it is important to estimate the correct factors with proper weights. The basic factors include credit payments history, time length of credit history, current debts and so on. Credit score is a crucial part of an individual's life as it allows banks and other financial organizations to anticipate if providing loan to a particular individual would be wise, and if it is, the level of trustworthiness can also be predicted. Therefore, it is a way to measure the risk associated with an individual. Credit score in the northern part of the world has become such an integral part of financial lives that provides a better understanding of how actions are affected by numbers. This permits people of any age with any income to benefit from receiving high credits. However, credit score is extremely fragile. It has the potential to harm an individual without being aware of it. This happens when wrong data is entered about an individual, which the individual is unaware of. For instance, such errors may cause an individual to not pay bills due to the wrong apartment number entered, leading to a low credit score in the perspective of the financial institution. Even though such problems are solvable, it is important to be aware of these errors for further harm.[48] Furthermore, the significance of using this score wisely plays a vital role in an individual's life. Actions and decisions should be taken carefully in order to maintain the score high for a better history. However, if the score decreases for any reason, there is no need to be devastated as the score is not a fixed value. This score will be persistently updated depending on the actions of the individual. Hence, the score is definitely improvable. Actions that increase the score depends on various parameters including paying back loans, bills on time with the correct amount, avoiding overextending of credit card, etc. To conclude, the importance and usefulness of credit scoring system is a necessity in financial areas as this is believed to improve any kind of losses by financial institutions. .[51]

## 1.3 Current Scenario & Motivation

The rise in the amount of non-performing loans and competition in the market of banking led to most commercial banks to intensely concentrate on credit risk assessment. As the chance of a borrower failing to reimburse the loan rises, the concept of credit risk is surfaced. Again, after the current financial predicament created from credit risk's poor administration, it has become a topic of great interest in the financial industry of Bangladesh.[22].Credit rating industry in Bangladesh started its journey in 2002. The process was initiated by Credit Rating Information Service Limited(CRISL) [52] as the primary registered credit rating system in Bangladesh. Credit Rating Agency of Bangladesh Limited (CRAB) was the second rating agency which went to operation on 2004. However,Bangladesh Bank circulated its Credit Risk Grading Manual in 2005. Initially, the Credit Risk Grading Manual was practiced to assess the grading of credit risk before banks allowed borrowers to lend. Whilst the reports of CRISL rating began to gain attention, Bangladesh Bank needed to be alerted about the valuable system of credit scoring, leading to persuading Bangladesh Bank to grab the ingenuity before the system becomes mandatory for public offering. It was concluded in the research CRA in Bangladesh [21] that although Bangladesh has numerous banks and financial institutions operating in a small economy, the Credit rating industry is yet to mature, given the financial turmoil and instability the Credit rating industry should be monitored strictly in order to ensure safety and impose adequate guidelines. It was noted that credit rating of specific individuals instead of organizations is not yet implemented properly in Bangladesh and there is also a scope for improvement in that sector. This will particularly benefit Micro-finance institutions like Brac micro-finance which provide loans in small amounts to individuals in need. And it is harder to predict the probability of defaulters at an individual level. Discussing the current scenario with the people working in Brac micro-finance it was visible that the current infrastructure of assessing an individual for eligibility has large room for improvement. The process is manual and involves tedious application process and field work. The process is lacking in accountability and transparency. There are no solid criteria or score for judging a new application. Hence, the implementation of a proper model to provide a quantitative score will help to strengthen the process. Our aim was to develop a robust model capable of assigning a quantitative score to an individual, given some financial history of that individual. This score would provide a basis of evaluation for that individual. We also wanted to ensure that the model is not treated as a 'black box' with no reasoning behind its output. Since credit worthiness is a sensitive issue determining the future of an individual, we tried our best to provide an interpretation of why the model is assigned a particular score, and which factor affected the result mostly.

## 1.4 Objectives

Loan defaulters cause an irreversible impact on both the country's economy and welfare of its financial organizations. The focus revolves around minimizing the number of debtors and forecast individual credibility. The objectives are as follows :

- Ease the trouble detecting defaulters of various financial institutions with the help of a credit score. This will allow banking systems and other financial organizations to reduce losses.
- Introduce a standardized measure of credit score of an individual. This score should be recognized by financial organizations and institutions among the country to better identify defaulters.
- This model will assist in removing the human bias, which is an extremely important factor in Bangladesh.
- The model will not completely automate the process, but provide a reference for human evaluation too.
- The model will provide a reason for the score. Therefore, in the case of confusion or unexpected results, the specific weights can be shown, providing the reason(s) for the specific result.

## 1.5 Challenges Faced

Initially, after being content with our research topic we got familiar with the process of credit risk assessment. Loan creditors, borrowers along with the technicalities associated with the process. We conducted a formal meeting with Brac Micro-finance department, which is the largest micro-finance institution in Bangladesh providing micro loans to both small and large businesses as well as individuals. They provided us insight about the evaluation process and management of micro-loans. We were supposed to conduct our analysis on data provided by Brac Micro-finance Institution department. Unfortunately, due to some technical difficulties involving lengthy digitization process of hard copy of application forms and availability of tabular data, the data-set has not been received.
However since we gained valuable intuition about the faults in the current system. We decided to take an approach to improve those using collected data set. Later if we are provided with real-world data, we can fit it to our developed model.
Computational complexity of performing both Grid-Search and Random-Search over the hyper-parameter space was also a hindrance due to the unavailability of powerful CPU or GPU. As conducting training and iteration on the models on an average CPU (Intel i5) and GPU (Nvidia 920mx) was extremely time consuming. Taking up at max 4-6 days performing grid search over the hyper-parameter space for a particular model.

## 1.6 Benefits of Automating the Credit Scoring Process

The rapid development of machine learning techniques has directed to an increased accuracy in the predictions from a large amount of data. In supervised learning, the input of the algorithm is a vector of features that outputs a classification. Be that as it may, the choices made by the model have the chance to be both positive or negative. For instance, if the predicting model alters the classification from high

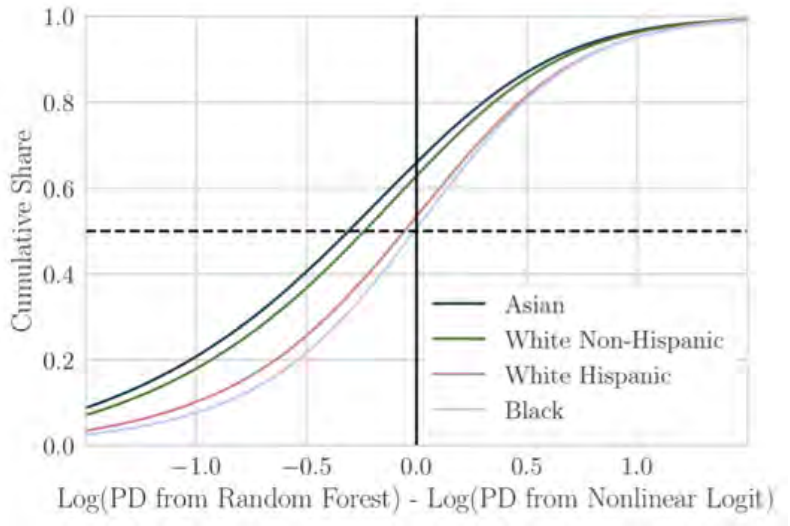


Figure 1.1: Credit Scoring Pattern with Ethnicity [38]

credit-worthy to low credit-worthy, the client may have a chance to be advertised less sums of advances. On the contrary, users who were previously offered less amount of loans could be offered more if the model learns that their true credit score was actually lower. Such assumptions may have distributional concerns. In the figure below, the horizontal axis represents the change in the log value that is used to predict the default probability as creditors move from traditional predictive models to machine learning models. The vertical axis represents the cumulative share of debtors experiencing changes. A variety of racial groups have been presented as debtors or borrowers.
The vertical solid line in the middle differentiates low-risk debtors marked by the machine learning models. From the figure, it is observed that 65% of Asian and Non-Hispanic borrowers are classed as less risky compared to 50% of Hispanic and black debtors. Therefore, the advances from newly machine learning models are skewed in the favor people enjoying the gain whereas disadvantaged groups are not as benefited. The reason for the indebted individuals to be assembled by race was not since the race was taken as input, but for the exceedingly non-linear combinations of a few variables. Consequently, as for an Asian nation like Bangladesh, it is more likely that the usage of modern machine learning models will exceedingly advantage the credit showcase.

## 1.7 Thesis Outline

The aim of this research was to construct a model capable of predicting a score based on an individual's financial history. As well as provide adequate reasoning behind the score. The aim of the authors was to formulate the best model for achieving this task and optimise it for accuracy. Along with it, provide visual reasoning behind the output.

To begin with, in the first chapter (Chapter 1), overview of credit score and its benefits in the financial sector are discussed. The Problem Statement was to develop and introduce this concept in context to our country, thereby help avoid millions

lost in default loans.

Secondly, in (Chapter 2) related work surrounding default loan classification and prediction are discussed. Outlining significant results achieved by researchers.As well as the lacking in existing methods were over viewed.

Next, in (Chapter 3) background analysis of various supervised algorithms are discussed along with their implementation details. These algorithms are used in the research pipeline later on.

In (Chapter 4) the Research Methodology and workflow is proposed. Details about data collection, processing, feature selection conducted in the research is elaborated.

Furthermore, in (Chapter 5) selected models are optimised by hyper-parameter tuning and Cross-validation to improve accuracy metric. Comparative Analysis were carried out between the various algorithms. A new model is proposed called RfDNN whose implementation and architecture is elaborated on. LIME analysis is also conducted on the optimized model and interpretation details are given.

Finally, in (Chapter 6) Experimental results and analysis is conducted. Visual comparisons of the models are provided. Accuracy metrics are tabulated. Proposed model RfDNN's capability to improve upon base Random Forest's accuracy is demonstrated. Conclusions and further work were drawn in the last chapter (Chapter 7)

# Chapter 2

# Literature Review

Credit risk assessment is a prominent matter in the field of banking and financing. Statistics and human evaluation are the key studies closely associated with it since it's instantiation. However recently due to the rapid advancements in data science and machine learning, credit risk assessment using Pattern recognition and Machine Learning have gained great significance in the research community. Plenty of noteworthy research papers have been published which gained traction in this area of study. Artificial Neural Networks have been used often in these papers. ANN, known to be an Artificial Neural Network, is a pattern for processing information that was developed enthused by the working mechanism of the biological nervous system. In [16] the authors used an RBF multilayer feed forward network, the results which were compared with a general logistic regression model. They concluded that the Logistic regression model had the upper hand when classifying positive classification whereas the Neural Network model had the edge when classifying negative applications. In [29] the authors also made a similar comparison, where they used the chi-square test to score the defaulters. Carrying out supervised training on a thousand instances, the logistic regression model outperformed the neural network. However, ensemble methods have received significant praise in the research community. The authors in [8] have used neural networks to build an ensemble agent where de-correlation maximization was used to choose the most suitable neural net models.The outputs from the models were integrated using different ensemble strategies-mean, median,max,min, product. To compare the ensemble approach, it was compared with single based agents(Logistic Regression, Support Vector Machines, and ANN), hybrid agents(Neuro-fuzzy, Fuzzy SVM) and a voting based reliability ensemble method.The reliability based neural network agent outclassed the other models marginally. It is also observed that Generalized Linear Modelling, distributed Random Forests and Gradient Boosting Method have also been implemented by many achieving noteworthy results. In [26] GBM was used on a Brazillian Bank Dataset. Generalized Linear Model and Random Forest were also used in this paper. 70% of over 20 thousand instances were used to train the models. GBM outperformed the aforementioned methods by a significant margin with an AUC Score of 99Classification and Regression trees (CART) commonly known as Decision Trees are used frequently in credit risk prediction. In [34] the authors performed a comparative analysis between tree based models and neural network approaches. In this proposal analysis was made between LogR, GBM, Random Forests and Neural network models.Modifications were conducted to reduce computational cost and alleviate

accuracy. The lambda and alpha hyper parameters of LogR was trained using Elastic Net to avoid "over-regularization". The number of trees of GBM and Random Forests were set to 120 . Lastly, four neural network models were implemented using different number of hidden layers, regularization functions. Grid Search was used to select optimum values of drop out ratio, activation functions, layers abd regularization functions.AUC Score and RMSE were used as metrics for evaluation. Results concluded tree based approaches, i.e Random Forests and GBM performed significantly better than the neural network and LogR models. Additionally, in the paper [11] the authors made a comparative analysis between models using decision trees, artificial neural networks, naïve bayes classifier, k-nearest neighbor classifier and a model based on linear discriminant analysis. Furthermore, ensemble models were made using these classifiers. It was observed that decision tree and the model based on naïve bayes classifier achieved the best results. It was also presented by the authors that it was difficult to find the best network topology for the neural network model and it was difficult to find the optimum value of k for the knn based model.

We found significant insight on the various difficulties and challenges associated with credit scoring from previous researches done on the topic. As mentioned in [23] it was found out that Parametric models like LDAs and Logistic regression were more accurate at predicting accurate overall score than neural network approaches such as Multi-layer Perceptron (MLP)[5] although Mixture-of-Experts (MOE)a modification of the MLP which decomposes the credit scoring task and assigns local experts to learn special parts of the problem, showed accuracy predicting bad credits comparable to Logistic Regression. This is mainly because in MOE architecture during back propagation affects the weights which are localized to the expert networks. Radial basis function networks (RBF) also stood out from other neural network approaches. We have also looked into [32] where they used Generalized Linear model algorithm, which is a modification of logistic regression model. Basically Logistic regression is a classification algorithm that generates a binary response when given a set of independent variables. GLM is an improvement of this method and provides a confidence bound where lies the probability of a positive outcome. Their model achieved Predictive confidence of 97.437%, while overall and average accuracy are over 98%. Further improvements were proposed by [44] who used Ensemble Logistic Regression boosted by GradientBoost[43] on German and Australian datasets found in UCI machine learning repository and acquired accuracies of 81% and 88.4% respectively. Taking into account all the findings we have decided to use an ensemble approach to our data mining problem.

# Chapter 3

# Background Analysis

## 3.1 General Supervised Algorithms

### 3.1.1 Linear Regression

Linear regression is a machine learning algorithm that follows supervised learning where a regression task is performed. Based on the independent variables, the model targets a prediction value, which can be used for forecasting and finding the relationship between variables.
Hypothesis function for linear regression is as follows :

$$Y = a \times \theta 1 + \theta 2$$

The variables x and y are given while the model is being trained.

- x: input training data (univariate – one input variable(parameter/s) or multivariate)
- y: labels to data (supervised learning)
- $\theta 1$: intercept
- $\theta 2$: coefficient of x

While the model is being trained, a best fit line is created that predicts the label (y values) for a given input value (x values). Hence, the best regression fit line is produced by finding the best possible values of $\theta 1$ and $\theta 2$.

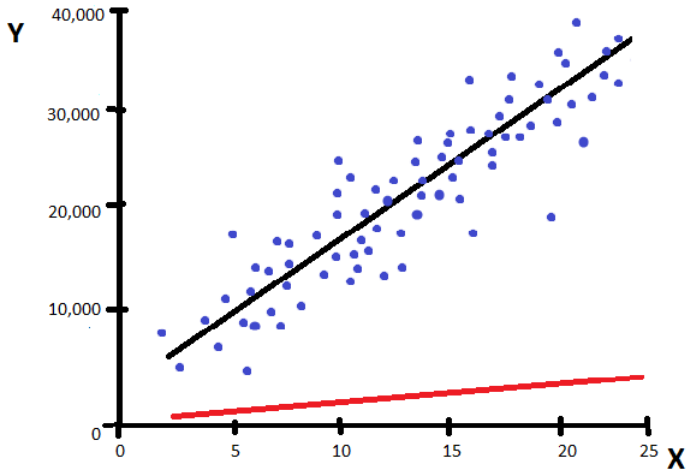


Figure 3.1: Linear Regression Intuition

**Training & loss**

Cost Function (J) : The goal of the model is to predict y values such that the error difference between the predicted value (y) and true value (x) is minimum. Therefore, updating the values of $\theta 1$ and $\theta 2$ is necessary in order to reach the least possible error.

$$minimize(\frac{1}{n}\sum_{i=1}^{n}(pred_i - y_i)^2)$$

$$J = \frac{1}{n}\sum_{i=1}^{n}(pred_i - y_i)^2$$

Hence, the cost function is the Mean Squared Error (MSE) between the predicted and true value.

**Gradient Descent:**
Gradient descent is used to achieve the best fit line by updating the values of $\theta 1$ and $\theta 2$ . This reduces the cost function and minimizes the MSE value. By beginning with random values of $\theta 1$ and $\theta 2$ , the model updates the values by iterating in order to achieve the minimum cost. The aim is to find the best suitable parameters $\theta 1$ and $\theta 2$.

### 3.1.2 K-Nearest Neighbor (KNN)

KNN is a non-parametric pattern recognition algorithm used for both classification or regression tasks. The intuition behind it is to find the k-closest training examples in the feature space, and the output depends on the property of those k-closest neighbours. For, classification it is classified by plurality vote of its neighbors. For regression, it is assigned the average value of its neighbours.
The k-NN algorithm is used for approximating the continuous variables, in a regression problem. In our task the algorithm works as follows:

- Compute the Euclidean distance from the training example to the labeled examples.

$$EuclideanDistance : \sqrt{\sum_{i=1}^{n}(q_i - p_i)^2}$$

.

- Order the labeled examples by increasing distance.
- Find a heuristically optimal number k of nearest neighbors, based on Root Mean Squared Error (RMSE).
- Calculate uniform weighted average with the k-nearest multivariate neighbors.

## 3.2 Ensemble Models

### 3.2.1 Decision Trees and Random Forests

#### Decision Trees

Decision Tree is a method that uses a flowchart-like tree structure, a collection of decisions and all of their possible results, including the input cost and utility.
Decision-tree is a supervised learning algorithm, which works for both continuous (regression) as well as categorical(Classification) output variables.

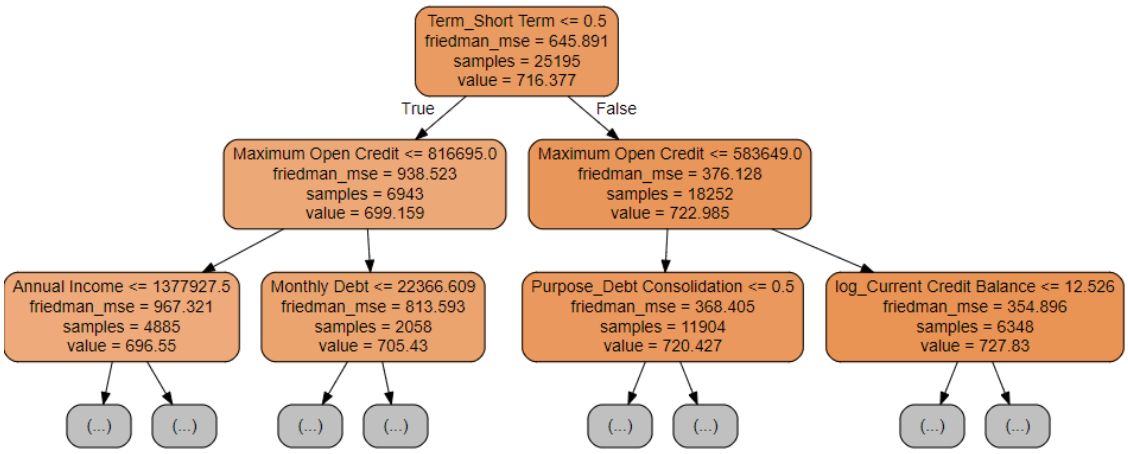


Figure 3.2: Decision Tree Path

An example 3.2 from our analysis, it can be visualized how decision trees make binary splits (Yes or No) on conditions to achieve the best split between the sample data given. The root node consists of all the input samples (x1, x2, . . . . . , xn). The samples are then split based on certain features that increases information gain.

- Each internal node of the tree is a feature/ attribute.
- Each leaf node corresponds to a label/prediction.

The core algorithm for building decision trees is called ID3 by J. R. Quinlan [2] which implements a top-down, greedy search through the space of possible branches with no backtracking. The ID3 algorithm can be used to construct a decision tree for regression by replacing Information Gain with Standard Deviation Reduction [53].
A decision tree is built top-down, partitioning the data into subsets that contain samples with similar values (homogenous). Standard deviation is used to calculate the homogeneity of a sample instance. A sample is completely homogeneous when its standard deviation is zero.

$$(StandardDeviation)\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2}, \quad \text{where} \quad (Mean)\mu = \frac{1}{N}\sum_{i=1}^{N} x_i$$

$$\text{CoefficientOfVariation(CV)} = (\delta/\mu) \times 100$$

Standard deviation for two variables (Target, Feature):

$$S(T, X) = \sum_{c \epsilon X} P(c)S(c)$$

The attribute with the largest standard deviation reduction is chosen for the decision node.
The standard deviation reduction relies on decreasing the standard deviation after a data-set is split on an attribute. To construct a decision tree it is required to find the feature that gives the highest standard deviation reduction (SDR) (i.e., the

most homogeneous branches). The feature corresponding to the highest standard deviation reduction (SDR) is selected for the decision node.

$$SDR(T, X) = S(T) - S(T, X)$$

The data-set is split based on the values of the selected attribute. The process is executed recursively on the non-leaf branches, until all of the samples in the dataset is processed.
Usually Coefficient of variation (CV) is chosen as the criterion for stopping the recursion, i.e if the CV falls below a certain threshold e.g 10% for a given branch then we stop the splitting process and assign the average value at the leaf node for that subset.

**Random Forests**

Random forests is a machine learning algorithm involving a large number of decision trees acting as an ensemble.The decision trees are generated by bagging and bootstrapping, i.e taking random subsets of the data-set to generate the trees with replacement. The individual trees produce a class prediction, of which the tree with the highest votes is highlighted as the model's prediction.
Random forests usually work excellently as a large number of uncorrelated trees are operating as a committee, outperforming any other individual constituent trees. The models or trees of random forests are observed to have low correlation, which is an advantage as low correlations clustering and binding together produce more accurate predictions than the sum of its individual predictions.The reason behind this theory is how excellently the trees shelter each other from their individual errors as numerous trees may be wrong or right, the following model will have a better chance to choose the correct path.[49]

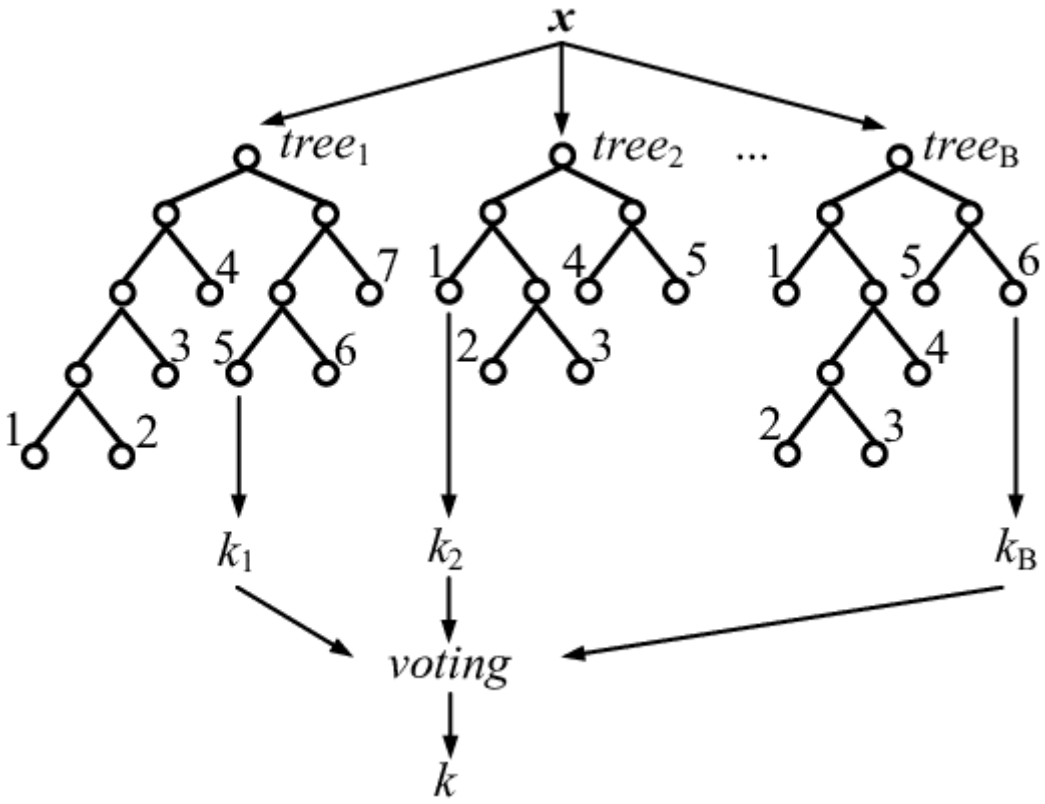


Figure 3.3: Random Forest Voting [17]

Therefore, the prerequisites of random forest to work excellently are as follows [17]:

1. Our features must have an actual signal for the model to work better at random guessing, meaning features should have some predictive power.

2. The errors and predicted results from individual trees should have low correlations. The features and hyper-parameters we choose will affect the correlations even though the algorithm itself attempts to engineer these correlations for us through feature randomness.

### 3.2.2 Gradient Boosted Regression

This is an ensemble method, i.e it uses the help of multiple predictors instead of just one to make the prediction. But instead of using Bagging technique like Random Forests, which build the decision tree predictors independently, GBR uses Boosting technique which generates the predictors sequentially. This technique employs the technique in which the following predictors learn from the mistakes of the previous predictors. So instead of relying on bootstrapping for choosing the samples, they are chosen based on errors made by the previous predictors. This results in faster convergence, close to actual predictions, but stopping criteria should be chosen wisely or it might lead to overfitting on the training data. GBR reduces variance and bias. [46]

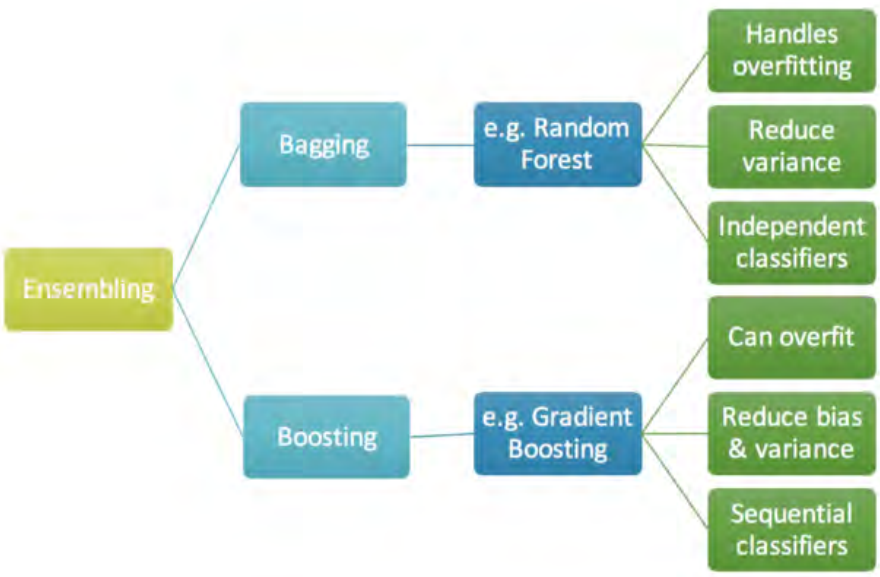


Figure 3.4: Ensembling Models [46]

The algorithm works by defining a loss function and iteratively using gradient descent to reduce the loss.
Loss Function:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

$$\hat{Y}_i = \hat{Y}_i + \alpha \times \delta \sum (Y_i - \hat{Y}_i)^2 / \delta \times \hat{Y}_i$$

$$\hat{Y}_i = \hat{Y}_i - \alpha \times 2 \sum (Y_i - \hat{Y}_i)^2$$

Where, $\alpha$ is the learning rate, and $\sum(Y_i - \hat{Y}_i)^2$ is the sum of residuals

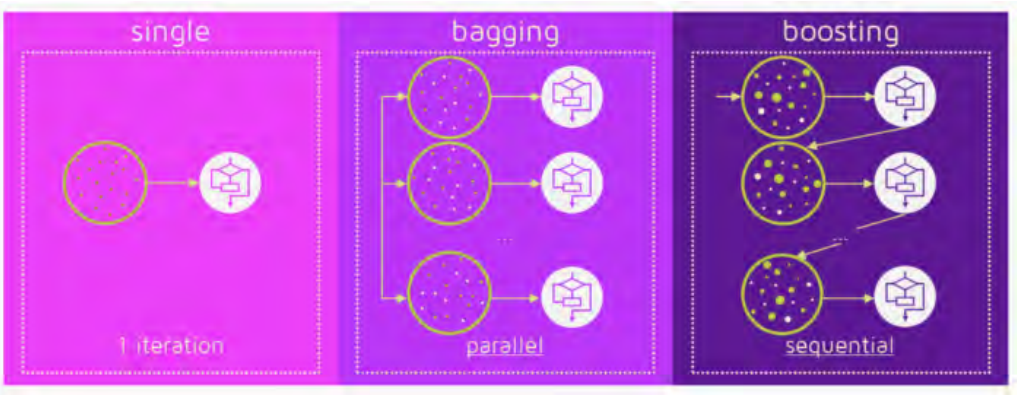


Figure 3.5: Difference Between Bagging and Boosting [33]

To simplify, we are basically updating the predictions such that the sum of our residuals is close to 0 (or minimum) and predicted values are sufficiently close to actual values. We run the algorithm for a number of fixed iterations.

### 3.2.3 XGBoost

XGBoost is a machine learning algorithm ensembling decision tree. It uses the framework of gradient boost. XGBoost is known to work for a wide range of problems including classification, regression, ranking and user-defined predictions. It is a portable algorithm as it has the ability to run smoothly on windows,Linux and OS X, and supports almost all major programming languages. Cloud interactions such as AWS, Azure and Yarn Clusters are also supported by XG Boost. The evolution of XG boost from decision trees has been shown below:

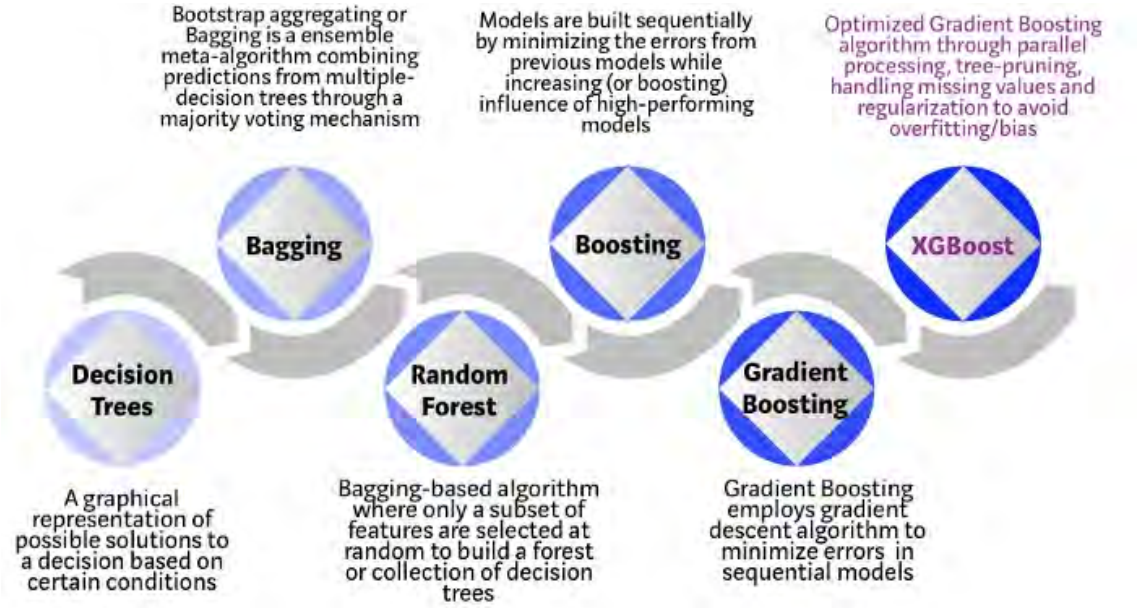


Figure 3.6: XGBoost Overview [50]

Gradient boosting machines are able to implement XGBoost with high scalability and accuracy. For boosted tree algorithms, it has the ability to push the limits of computing power . The aim of building this algorithm was for model performance and computational speed, especially engineered to exploit each bit of memory and resources of hardware. XGBoost can handle missing values, block structure to support parallelization for tree construction and can excellently fit and boost on new data that has been added to the training model. Tianqi Chen, the developer of XGBoost, believes this algorithm uses a more regularized model formalization in order to control over-fitting, resulting in better performance.

# Chapter 4

# Research Methodology

## 4.1 Dataset

Even though the data was supposed to be retrieved from Brac Microfinance, due to unfortunate circumstances and shortage of data, the actual data has not yet been received. Therefore, to conduct our data analysis we opted for a large enough reliable dataset [31] with 100514 entries extracted from kaggle.

## 4.2 Project Work-flow

The work-flow consists of using the provided bank loan data to develop a model that can predict an individual's credit score, and then interpret the results to find the variables that are most predictive of the score. This is a supervised, regression machine learning problem: given a set of data(x) with targets(y) (in this case the credit score) included, we want to train a model that can learn to map the features (also known as the explanatory variables) to the target.

- Supervised problem: we are given both the features and the target
- Regression problem: the target is a continuous variable (credit score is a number between 0-800) Machine Learning Work-flow

Although the exact implementation details can vary, the general structure of a machine learning project stays relatively constant:

- Data cleaning and formatting
- Exploratory data analysis
- Feature engineering and selection
- Establish a baseline and compare several machine learning models on a performance metric
- Perform hyperparameter tuning on the best model to optimize it for the problem
- Evaluate the best model on the testing set

- Interpret the model results to the extent possible

Draw conclusions and write a well-documented report This is not a linear workflow, as in sequence is not always maintained in the pipeline. A step can be visited more than once based on evaluation further down the pipeline. It is an iterative process

| | | |
|---|---|---|
| Loan ID | 100000 | non-null object |
| Customer ID | 100000 | non-null object |
| Loan Status | 100000 | non-null object |
| Current Loan Amount | 100000 | non-null float64 |
| Term | 100000 | non-null object |
| Credit Score | 80846 | non-null float64 |
| Annual Income | 80846 | non-null float64 |
| Years in current job | 95778 | non-null object |
| Home Ownership | 100000 | non-null object |
| Purpose | 100000 | non-null object |
| Monthly Debt | 100000 | non-null float64 |
| Years of Credit History | 100000 | non-null float64 |
| Months since last delinquent | 46859 | non-null float64 |
| Number of Open Accounts | 100000 | non-null float64 |
| Number of Credit Problems | 100000 | non-null float64 |
| Current Credit Balance | 100000 | non-null float64 |
| Maximum Open Credit | 99998 | non-null float64 |
| Bankruptcies | 99796 | non-null float64 |
| Tax Liens | 99990 | non-null float64 |

Table 4.1: List of initial columns in data-set

## 4.3 Data Pre-Processing

The initial concern was how to deal with the missing values in the dataset. Following is the initial percentages of missing values by columns 4.1.

| | Missing Values | % of Total Values |
|---|---|---|
| Months since last delinquent | 53655 | 53.4 |
| Credit Score | 19668 | 19.6 |
| Annual Income | 19668 | 19.6 |
| Years in current job | 4736 | 4.7 |
| Bankruptcies | 718 | 0.7 |
| Tax Liens | 524 | 0.5 |
| Maximum Open Credit | 516 | 0.5 |
| Current Credit Balance | 514 | 0.5 |
| Number of Credit Problems | 514 | 0.5 |
| Number of Open Accounts | 514 | 0.5 |
| Loan Status | 514 | 0.5 |
| Years of Credit History | 514 | 0.5 |
| Current Loan Amount | 514 | 0.5 |
| Purpose | 514 | 0.5 |
| Home Ownership | 514 | 0.5 |
| Term | 514 | 0.5 |
| Monthly Debt | 514 | 0.5 |

Figure 4.1: Initial missing value percentages of each column in the dataset.

'Months since last delinquent' column with more than 50% missing data was dropped because processing it won't be reliable. The last 514 entries of the dataset was null so they had to be dropped. The rest of the missing data of the numeric columns were filled with the mean of all the entries of that column. The categorical variables were filled up randomly, as there were very few missing values in them.
Box Plots of the columns were used to detect outliers, if there were any present in the dataset. An example of an outlier in annual income column is given in 4.2

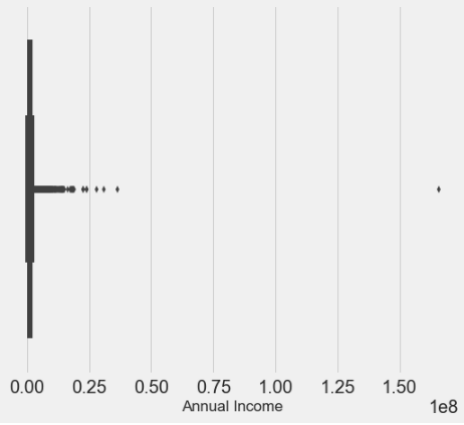


Figure 4.2: Box-Plot of 'Annual Income' column

The entry on the far right of the plot is an example of an outlier which had to be dropped due its large deviation from the median.

The skewness of distributions of the columns were observed to get an insight into the data-set.
Feature scaling was conducted to normalize the numeric data. Min-max strategy was used. Which is denoted by the formula

$$z = \frac{x - min(x)}{max(x) - min(x)}$$

## 4.4 Feature Selection & Engineering

To figure out relation of categorical values, the credit score Density plots were plotted for each categorical features.
The plots showed the effect of different categorical feature values on the credit score and how the density distribution varied with them 4.3. Features whose value did not affect the distribution of the credit score was observed so that in the final model they could be given less priority.

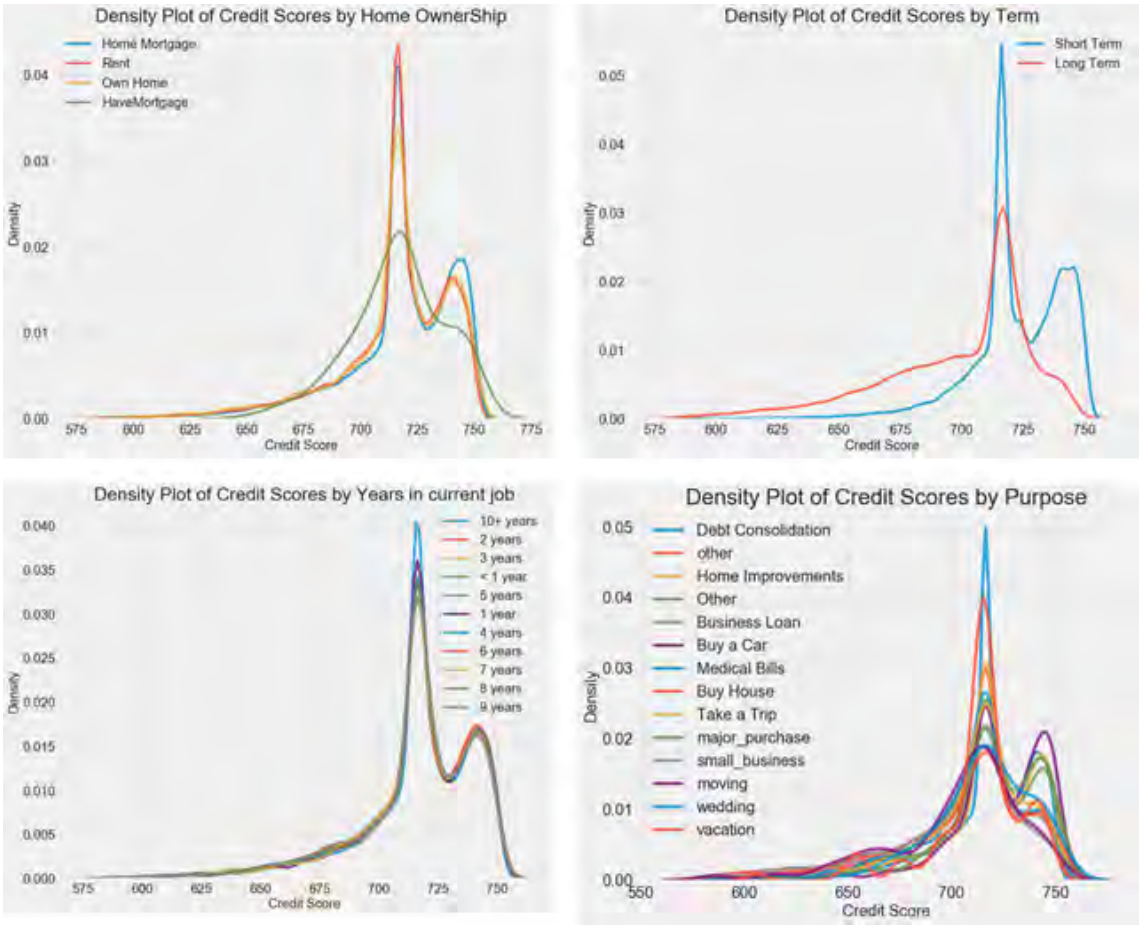


Figure 4.3: Density Plots of Categorical features against Credit Score.

It could be seen that except 'Years in current Job', the rest of the categorical features affected the distribution of the credit score. Not including 'Years in current Job' creates no issue in the final model, as change in its value does not affect the distribution of the Credit Score. It can also be observed from the distributions

that they are bimodal. In statistics, a bimodal distribution is a continuous probability distribution with two different modes. These appear as distinct peaks (local maxima) in the probability density function [22].

- Feature Selection: The process of selecting the features with the most relevancy in the data, depending on various factors. High relevant features could be a feature with the greatest correlation or highest variance with the target. Thus, less relevant features are removed in order to assist the model to better generalize and understand the new data.

- Feature Engineering: The method of selecting the raw data and extract, also known as forming new features that allow machine learning models to absorb a mapping of features with the targets. Transformation of variables are usually used including logarithms and square roots. However, categorical variables of one-hot encoding may be used. Therefore, the process of feature engineering summarizes to deriving extra features that are relevant from the raw data.

Feature engineering and selection are iterative processes that will usually require several attempts to get right. Often we will use the results of modeling, such as the feature importance from a random forest, to go back and redo feature selection, or we might later discover relationships that necessitate creating new variables hence requiring feature engineering. Moreover, these processes usually incorporate a mixture of domain knowledge and statistical qualities of the data.
After carrying out Feature Engineering by introducing the log and square roots of numeric columns, we needed to remove multi-linearity. i.e. Finding features which are highly col-linear due to some underlying similarity, hence keeping them is redundant. Features with col-linearity between them above a certain threshold (0.65) was removed from the feature set. Heat-map illustrating the correlations between the features were generated to find out multi-linearity or collinearity. 4.4

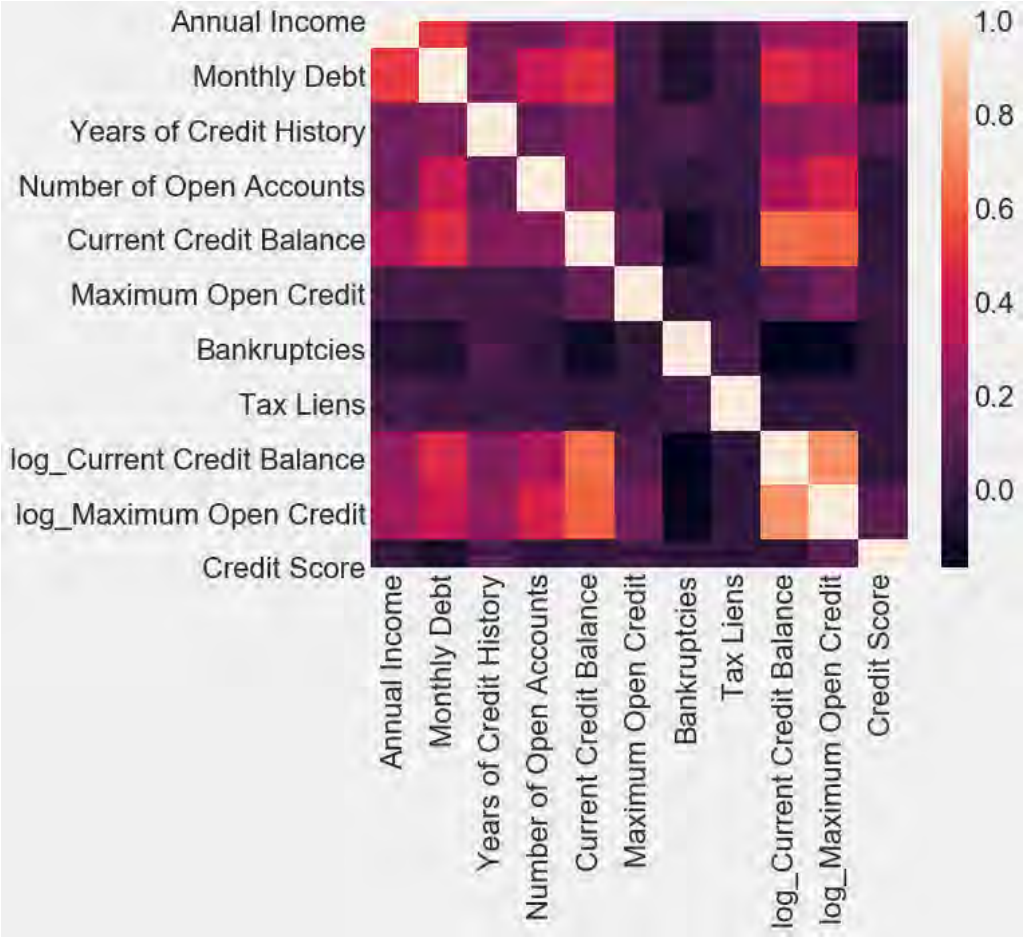


Figure 4.4: Heat-map of Numeric Features.

After conducting feature selection and engineering we ended up with the following feature set:

| | | |
|---|---|---|
| Current Loan Amount | float64 | Categorical values are one hot encoded |
| Annual Income | float64 | Loan Status |
| Monthly Debt | float64 | Term |
| Years of Credit History | float64 | Home Ownership |
| Number of Open Account | float64 | Years in current job |
| Number of Credit Problem | float64 | Purpose |
| Current Credit Balance | float64 | |
| Maximum Open Credit | float64 | |
| Bankruptcies | float64 | |
| log MonthlyDebt | float64 | |
| log MaximumOpenCredit | float64 | |
| sqrt TaxLiens | float64 | |

Table 4.2: Final Set of Features selected

## 4.5 Train-Test Split

The features(X) and Target(Y= "Credit Score") were separated and Train Test split was carried out. With 30% in test set and 70% in the training set. The training set would be used to train our model. And accuracy evaluation would be done using the test set, which is unseen data for the trained model.
A Baseline prediction was done using the median of the training set:
The baseline guess is a score of 716.28 Baseline Performance on the test set: Mean Absolute error = 17.6026.

# Chapter 5

# Model Implementation and Optimization

## 5.1 Work Flow Overview

It is important to use the correct model space, given the training set in order to have the best possible model. Therefore, our primary goals are to aim towards minimizing the true error on the test set and to avoid over-fitting the data on the training set. Following is an overview of our work-flow 5.1.
The details are as follows:

- Raw Data-set is Cleaned and Processed. This includes imputing missing values, identifying outliers and feature scaling
- Feature Selection and Engineering is carried out. Selection of important features using correlation and feature importance as reference. performing Recursive Feature importance Elimination. Feature Engineering is conducted by adding square root and logs of numeric columns and removing multi-linearity in the features.
- Train Test split carried out in the ratio 70:30, where the test set is held out for evaluation in final model.
- Randomized Search over hyper-parameters of selected models are conducted to further filter models and parameters to conduct Grid Search on.
- Models are optimized using Cross-Validation along with Randomized and Grid Search over the hyper-parameter space
- Final model evaluated on hold out test set. Results obtained and analyzed. Comparative analysis conducted between the models and the final model.

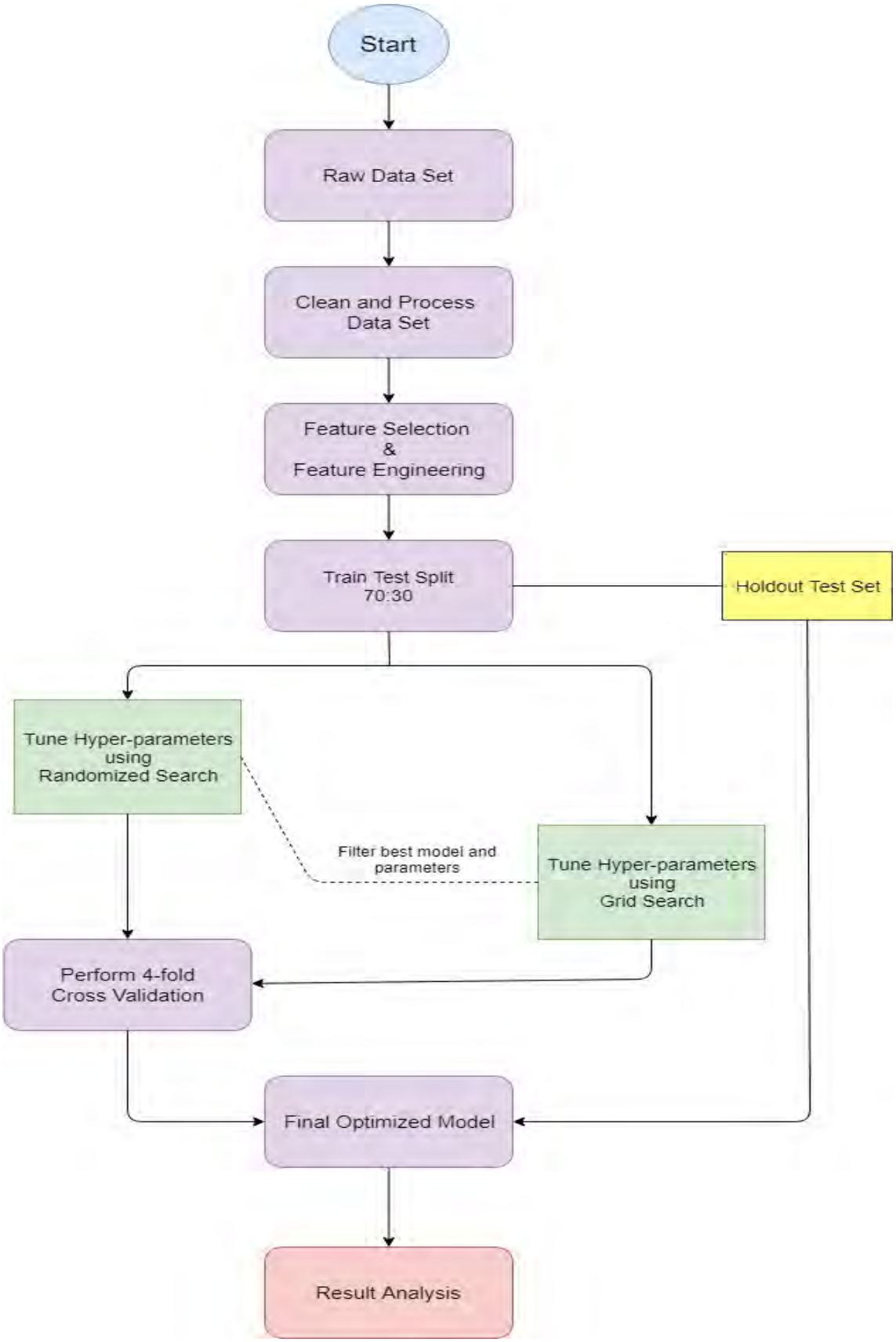


Figure 5.1: Proposed Work-flow

## 5.2 Evaluating and Comparing Machine Learning Models

Build, train and evaluate several machine learning methods for our supervised regression task. The objective is to determine which model holds the most promise for further development (such as hyper parameter tuning). We are comparing models using the mean absolute error. A baseline model that guessed the median value of the score was off by 17.6.

*Scaling Features*
This is necessary because features are in different units, and we want to normalize the features so the units do not affect the algorithm. Linear Regression and Random Forest do not require feature scaling, but other methods, such as support vector machines and k nearest neighbors, do require it because they take into account the Euclidean distance between observations. For this reason, it is a best practice to scale features when we are comparing multiple algorithms [23].
There are two ways to scale features:

- For each value, subtract the mean of the feature and divide by the standard deviation of the feature. This is known as standardization and results in each feature having a mean of 0 and a standard deviation of 1.
- For each value, subtract the minimum value of the feature and divide by the maximum minus the minimum for the feature (the range). This assures that all the values for a feature are between 0 and 1 and is called scaling to a range or normalization.

Both the test and training sets were scaled and normalized. Five different machine learning models were trained and evaluated using the great Scikit-Learn library.

1. Linear Regression
2. Support Vector Machine Regression
3. Random Forest Regression
4. Gradient Boosting Regression
5. K-Nearest Neighbors Regression

The default models were trained on the training set and their mean absolute error on the test set was calculated 5.2.

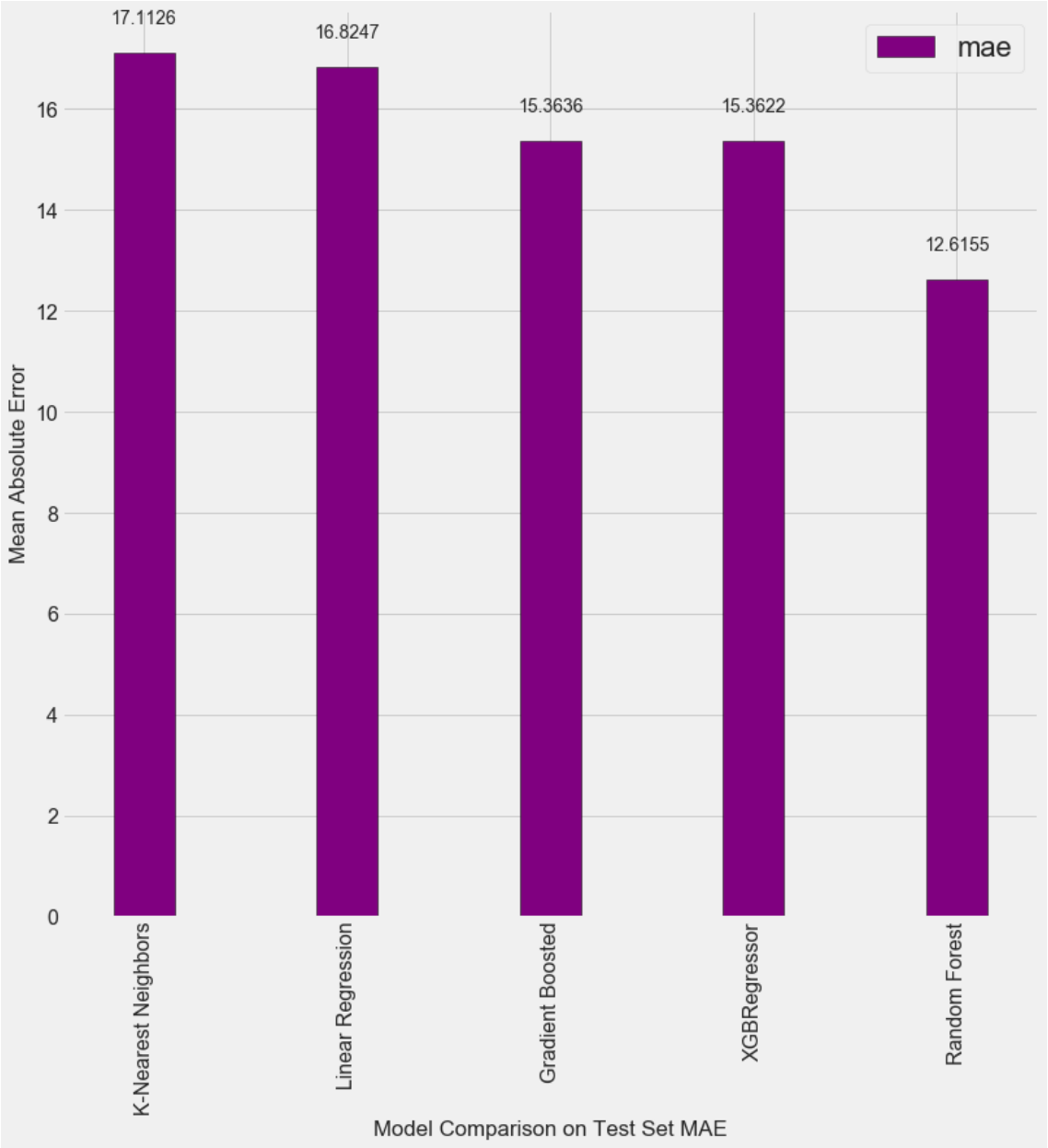


Figure 5.2: Baseline model MAE on Test Set

Although this is not a fair comparison because all the default parameters are used in making the models, but from the errors it can be inferred that the problem is Learnable, because all the models did significantly better than the baseline MAE of 17.62.

## 5.3 Model Optimization

In machine learning, optimizing a model means finding the best set of hyperparameters for a particular problem. The difference between model hyperparameters and model parameters are [24]

- Model hyperparameters are best thought of as settings for a machine learning algorithm that are tuned by the data scientist before training. Examples would

be the number of trees in the random forest, or the number of neighbors used in K Nearest Neighbors Regression [41].

- Model parameters are what the model learns during training, such as the weights in the linear regression.

## 5.4 Hyperparameter Tuning

We can choose the best hyper-parameters for a model through random search and cross validation.

- Random search refers to the method in which we choose hyper parameters to evaluate: we define a range of options, and then randomly select combinations to try. This is in contrast to grid search which evaluates every single combination we specify. Generally, random search is better when we have limited knowledge of the best model hyperparameters and we can use random search to narrow down the options and then use grid search with a more limited range of options so that specific hyperparameters can be tuned with a finer precision [41].

- Cross validation 5.3 is the method used to assess the performance of the hyper parameters. Rather than splitting the training set up into separate training and validation sets which reduces the amount of training data we can use, we use K-Fold Cross Validation. This means dividing the training data into K folds, and then going through an iterative process where we first train on K-1 of the folds and then evaluate performance on the kth fold. We repeat this process K times so eventually we will have tested on every example in the training data with the key that each iteration we are testing on data that we did not train on. At the end of K-fold cross validation, we take the average error on each of the K iterations as the final performance measure and then train the model on all the training data at once. The performance we record is then used to compare different combinations of hyper-parameters [41].

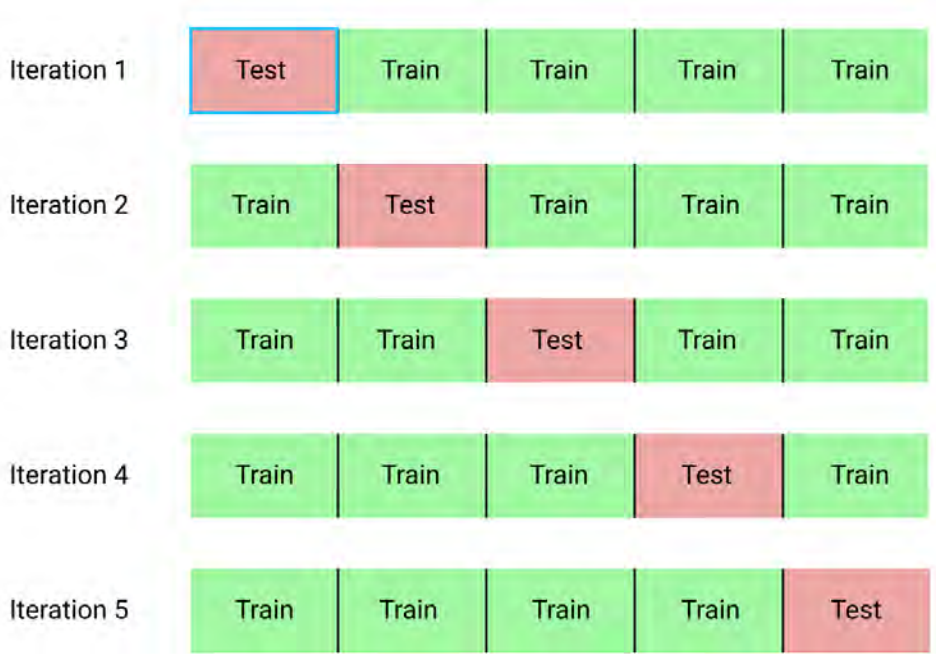


Figure 5.3: Cross-Validation Overview [41]

We chose the best two candidates from the default model test for model optimization.

- Random Forest Regression
- Gradient Boosting Regression

Both of these algorithms make use of generating regression and decision trees, based on information entropy of the features. A brief comparison of their characteristics [25]:

- Boosting is based on weak learners (high bias, low variance). In terms of decision trees, weak learners are shallow trees. Boosting reduces error mainly by reducing bias, i.e. the error for the wrongful assumptions we make building the learning algorithm. It is the primary reason for under fitting the model.Boosting runs sequentially hence parallel processing power cannot be used and its run-time is slower[1].
- On the other hand, Random Forest uses fully grown decision trees (low bias, high variance). It tackles the error reduction task in the opposite way: by reducing variance. The trees are made uncorrelated to maximize the decrease in variance, but the algorithm cannot reduce bias (which is slightly higher than the bias of an individual tree in the forest). Hence the need for large, unpruned trees, so that the bias is initially as low as possible. Random Forests generates trees in parallel, so runtime is faster[1].

After performing Randomized search of the following hyper-parameters 5.4 on both the algorithms and Cross validation, it was found out that Gradient boosting regression outperformed Random forests in terms of mean absolute error 5.5.

```python
# Loss function to be optimized
loss = ['ls', 'lad', 'huber']

# Number of trees used in the boosting process
n_estimators = [100, 500, 900, 1100, 1500]

# Maximum depth of each tree
max_depth = [2, 3, 5, 10, 15]

# Minimum number of samples per leaf
min_samples_leaf = [1, 2, 4, 6, 8]

# Minimum number of samples to split a node
min_samples_split = [2, 4, 6, 10]

# Maximum number of features to consider for making splits
max_features = ['auto', 'sqrt', 'log2', None]

# Define the grid of hyperparameters to search
hyperparameter_grid = {'loss': loss,
                       'n_estimators': n_estimators,
                       'max_depth': max_depth,
                       'min_samples_leaf': min_samples_leaf,
                       'min_samples_split': min_samples_split,
                       'max_features': max_features}
```

Figure 5.4: List of Hyper-parameters tuned

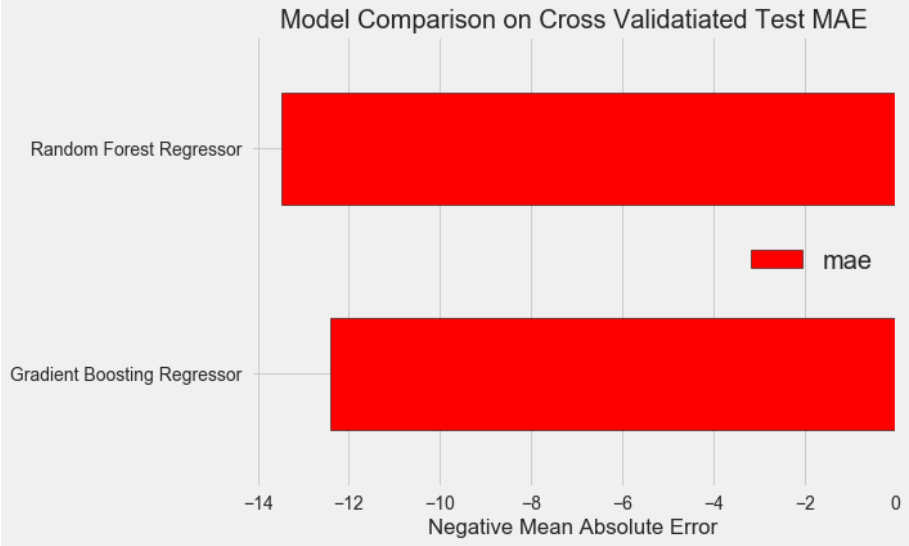


Figure 5.5: Error Comparison of best estimators predicted after Randomized Search and Cross Validation.

A grid search, which is a complete search using all the parameters and not just randomly chosen ones, was performed for number of trees on the best estimator of GBR. To figure out the effect number of trees used in making the model has on reducing training and test error 5.6.

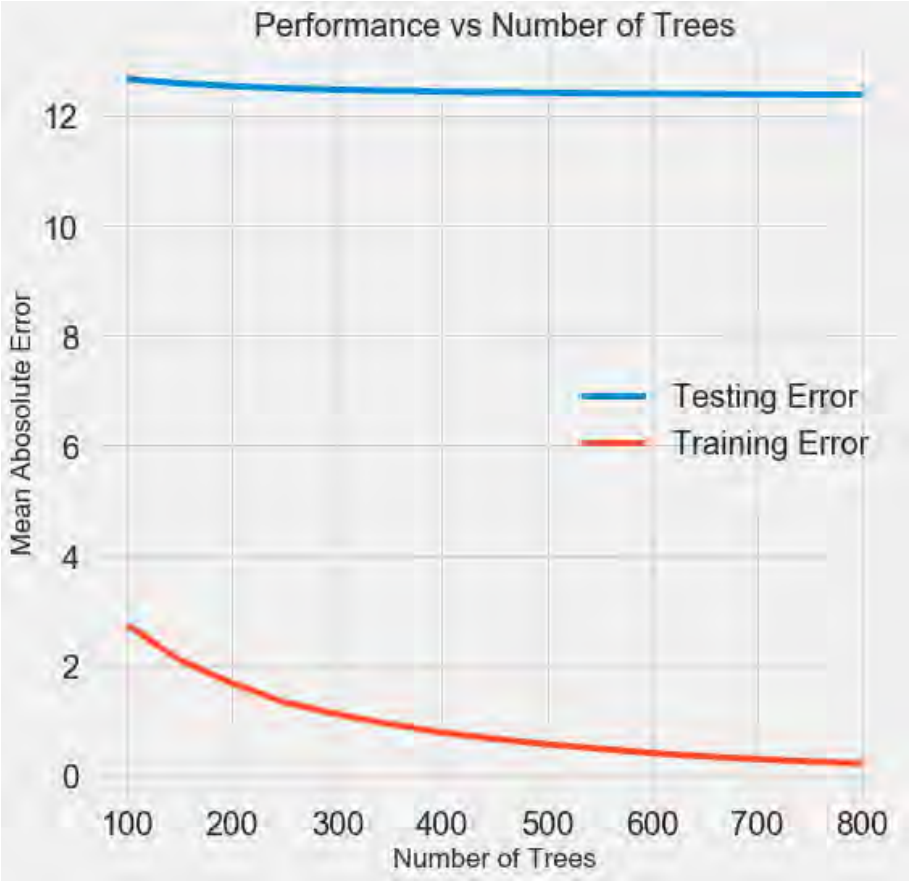


Figure 5.6: Effect of Number of trees on train and test error.

It is observed that increasing the number of trees is reducing the training error, but has negligible impact on the test error after 200 trees.
So, the model is overfitting on the training data with increasing tree number used. So we decided to limit the number of trees used while fitting our model to 200.

## 5.5 Proposed Hybrid-Stacked Model(RfDNN)

### 5.5.1 Overview

The intuition behind this model is to make use of a feature detector which would be the input downstream down a Neural Network. Implementing a supervised feature detector on top of DNN architecture as carried out by Yunchuan Kong & Tianwei Yu in their paper [40] was the motivation behind this method. Literature shows that among the machine learning techniques, random forests[6] (RF) have been an outstanding performer in learning feature representations [13],[19], given their robust classification power and easily interpretable learning mechanism
We decided to use the Random Forest model we optimized by tuning, use it as a "feature detector" and use it as an input to the downstream neural network model. Random Forest was a good choice for this methodology instead of Gradient Boosted Regression was because the Decision trees in Random Forest are trained independently from bootstrapping, whereas GBR trains the Decision trees sequentially minimizing the error incurred by the following trees. So the output from individual decision trees of a Random Forest provides an unbiased feature representation, which could be used as an input to a neural network to train on.
Our proposed model follows the Forest Deep Neural Network(fDNN) architecture mentioned by Yunchuan Kong & Tianwei Yu in their paper[40]
The following flowchart summarizes the architecture of our RfDNN model5.7:

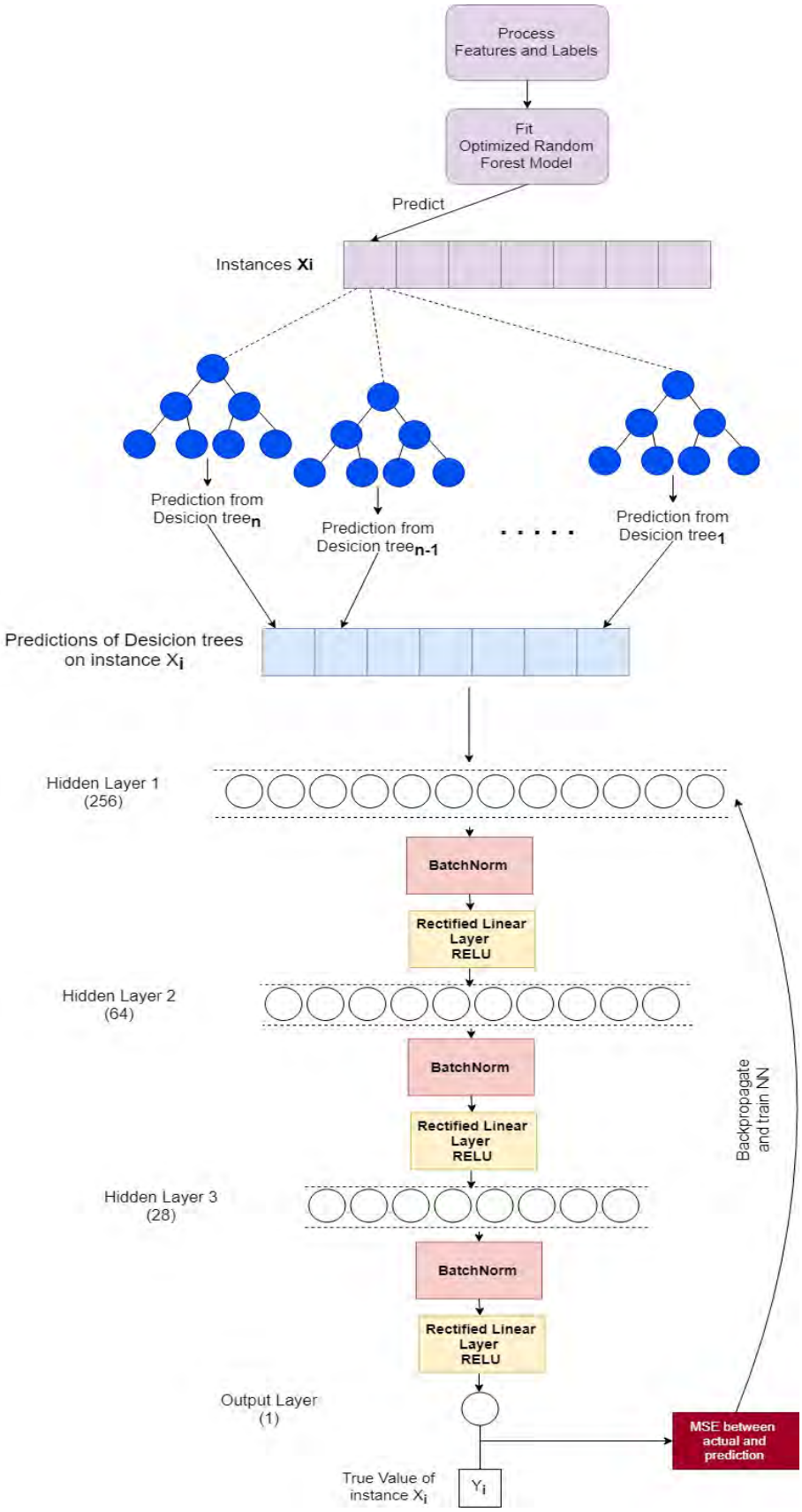


Figure 5.7: RfDNN model Architecture

### 5.5.2 Training and Architecture details

The training of RfDNN regressor consists of two steps. In the first step, selected and processed features are used to fit the Optimised Random Forest Model, then predictions from each individual decision tree of the forest for all the instances are recorded and then fed into the fully-connected DNN, for training in the second stage. After the two-stage training on training data, given a testing instance during inference time, the prediction is calculated using the entire model by the fitted forest and DNN.

The Deep Neural Network architecture consists of three hidden layers of size 256, 64, 28 and finally the output layer generating one value, the credit score. The Mean Squared Error between this prediction $\hat{Y}$ and True value $Y$ for the instance Xi is calculated and this is back-propagated through the network to train the model.
The activation function used in the model is Rectified Linear Unit (RELU)[10] with the form:

$$\sigma_{ReLU}(x) = max(x, 0).$$

This activation function has a benefit over Sigmoid activation and tanh activation over the fact that it reduces the problem of vanishing gradients during backpropagation through the model[7].
For regularization we used Batch Normalization [20] between the layers to reduce co-variate shift in the hidden unit values. This also prevents over-fitting as it has slight regularization effect.
For the model's optimizer we chose Adam optimizer[18], as it is the most widely used variant of Gradient Descent Algorithms used in Deep Learning research nowadays. We also make use of the mini-batch training strategy by which the optimizer randomly trains a small proportion of the samples in each iteration[24]. In this two step model, a variety of hyper-parameters need to be taken into account. The hyper-parameters of the Random Forest model were already optimized using Randomized and Grid Search. Hyper-parameters associated with the neural network consisting of the learning rate of the optimizer, beta and gamma parameters of Batch Normalization layer, number of epochs of training, were tuned using Randomized Search over a fixed range. The model is implemented in Python with packages Scikit-learn[12] and Pytorch. The proposed model performed marginally better in terms of Mean Absolute Error (MAE) than the base model consisting of a single random forest prediction.

## 5.6 Model Interpretability

After carrying out necessary hyper-parameters tuning and optimization of the ensemble models, they achieved an impressive accuracy on the test data-set ( 93%). Although this might be enough for certain problem domains, that is not the case for the task of credit scoring an individual. Treating the model as a 'black-box' with no reasoning behind the output raises both ethical and reliability issues. The way these algorithms are designed demonstrates there is no straight-forward path to determine why or how the output was generated. However, in recent research

focus has been shifted to understand these 'black-box' models better and significant progress has been made to provide reliable information on dynamics and the relationship between input, output and intermediates. As described in the paper [27], traditional methods of dimension reduction and Principal Component Analysis(PCA) are being improved upon. Research is being continued to provide better visual artifacts for further in-depth understanding. Moreover, different models may have different methods to analyzing. Despite the observations made, they might not directly correspond to trustworthy results [15] This paper summarises up the importance and failings in interpreting models and the different methods currently used to best capture the underlying mechanics of machine learning models.
Model Agnostic [25] This paper illustrates the benefits of using a model agnostic approach to interpret an ML model where an interpret-able method is generated from the predictions of the 'black box model'. Hence it is not model-dependent and can provide interpretations of more complex models such as Deep Neural Networks. This lets practitioners be more flexible and not rely on only traditional interpret-able models such as Linear or Logistic Regression.
Therefore, we have decided to conduct two Model-Agnostic methods to visualize our model output.Agnostic model methods are those that are not model dependent and works well despite the algorithm is used.

- Local Interpretable Model-Agnostic Explanations (LIME) [28]
- Feature Importance
- Single Decision Tree interpretation

## 5.7 Feature Importance

The concept behind calculating feature importance is fairly simple yet effective. It states that the importance of a particular feature in a data-set directly proportional to the increase in the prediction error of the model after we permuted or shuffled the feature's values, which breaks the association between the feature and the true outcome.
To measure the importance of a feature we calculate the increase in the model's prediction error after permuting the feature. A feature is only "important" if shuffling its values increases the model error, hence in this case the model relied on the feature for the prediction. Alternatively, a feature is not that important if shuffling its values caused little to no change in the model's prediction accuracy. The shuffling feature importance measurement was introduced by [6] for random forests. Based on this concept, [37] proposed a model-agnostic version of the feature importance and called it model reliance. Input: Trained model f, feature matrix X, target vector y, error measure L(y,f).
The algorithm used is as follows [47]:

1. Estimate the original model error eorig = L(y, f(X)) (e.g. mean squared error)
2. For each feature j = 1,. . . ,p do:
    - Generate feature matrix Xperm by shuffling feature j in the data X. This breaks the association between feature j and true outcome y.
    - Estimate error eperm = L(Y,f(Xperm)) based on the predictions of the permuted data.
    - Calculate permutation feature importance FIj= eperm/eorig. Alternatively, the difference can be used: FIj = eperm - eorig
3. Sort features by descending FI.

We carried out feature importance of our optimized models on the valid set predictions. And listed down the top 8 features according to their importance.

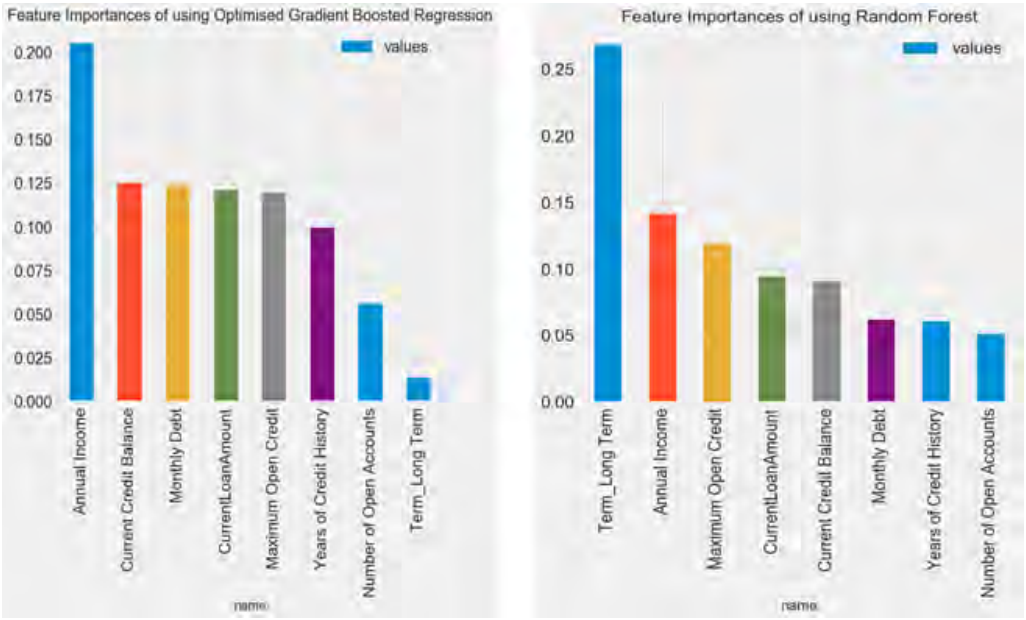


Figure 5.8: Feature Importance of RF and GBR models

It was observed that for both Random Forest and Gradient Boosted Regression models, annual income, maximum open credit, current credit balance, monthly debt and current loan amount significantly affected the predictions of the models. It would also be safe to assume that these factors are in fact crucially important when considering an individual for manual evaluation. Hence, our models seem to be prioritising the correct features when making a prediction, similar to what a human evaluator would predict. However, to strengthen our claim we conducted further analysis on both the model predictions.

## 5.8 Local Interpretable Model-Agnostic Explanations

The following paper impressively explains an individual prediction of an ML model proposed by [28]. We believe it strongly fulfills the three basic requirements for model interpret-ability.

- Model Agnostic: It is not model-dependant. Draws conclusion from only perturbing the input and predict behaviour from how the prediction changes.
- Interpret-ability: Explanations has to be user friendly to understand and this may become a constraint even for linear models containing large number of features creating a complex feature space. LIME's explanations use a data representation (called interpret-able representation) that is different from the original feature space. [36]
- Locality. LIME produces an explanation by approximating the black-box model by an interpret-able model (for example, a linear model with a few non-zero coefficients) in the neighborhood of the instance we want to explain.

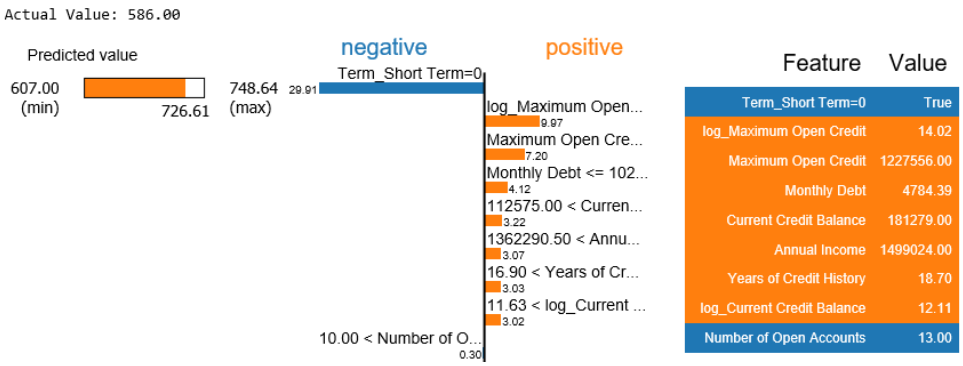


Figure 5.9: Wrong Prediction Interpretation

The term or duration of the loan seemed to affect the wrong prediction by a significant amount, followed by Maximum Open Credit. Monthly debt, Current Credit Balance and Annual Income.

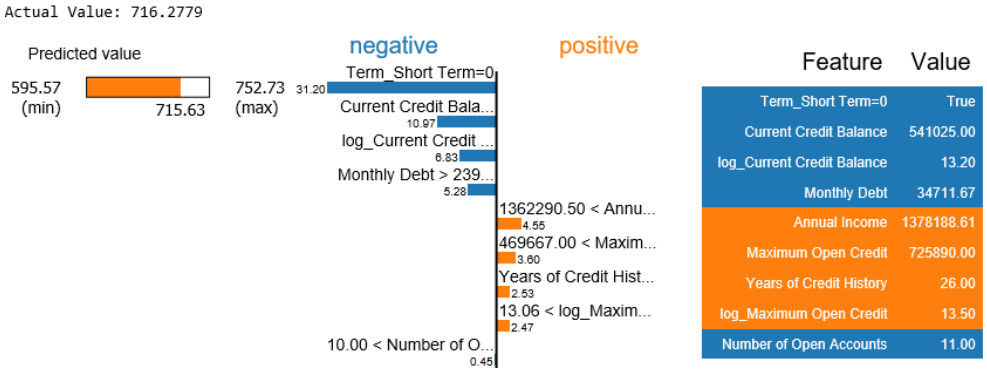


Figure 5.10: Right Prediction Interpretation

In the correct prediction 5.10 made by the model, it can be seen that the negative impacts on the prediction are due to high values of current credit balance & monthly debt and these are in fact considered red flags when evaluating a loan request. The duration being long term also negatively affects the prediction and this is logical given that long term loans are at higher risk of defaulting while reasonable salary and long credit/loan history favor the decision of the prediction. Given these interpretations, we can be further confident on our model in choosing the correct features to produce positive and negative weights during a prediction.

## 5.9 Single Decision Tree Interpretation

Lastly, we decided to use a model specific interpretation. Since both Random Forest and Gradient Boosted Trees use decision trees as individual predictors, it was necessary to further investigate a single decision tree in order to interpret the model better and gain an insight into how and where the splits were made when making a prediction.

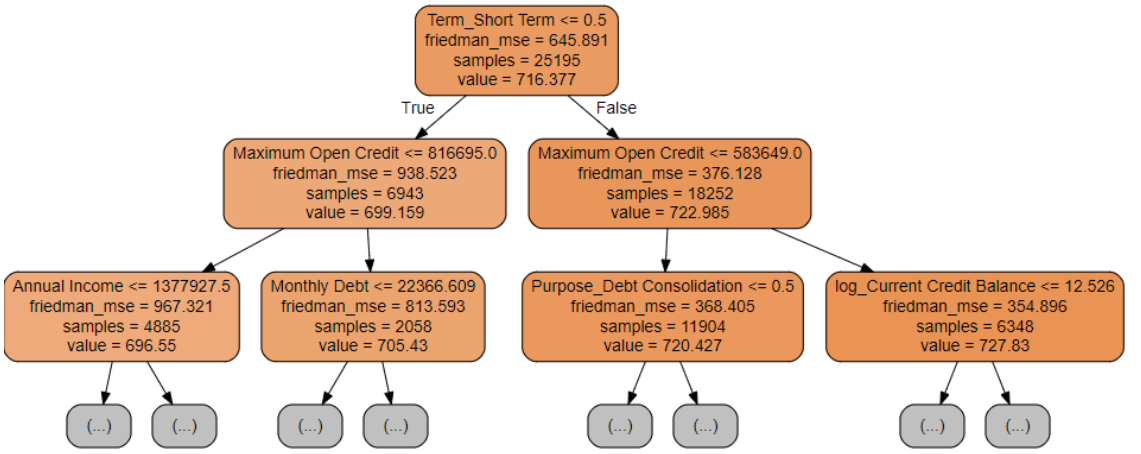


Figure 5.11: Tree Interpretation of RF

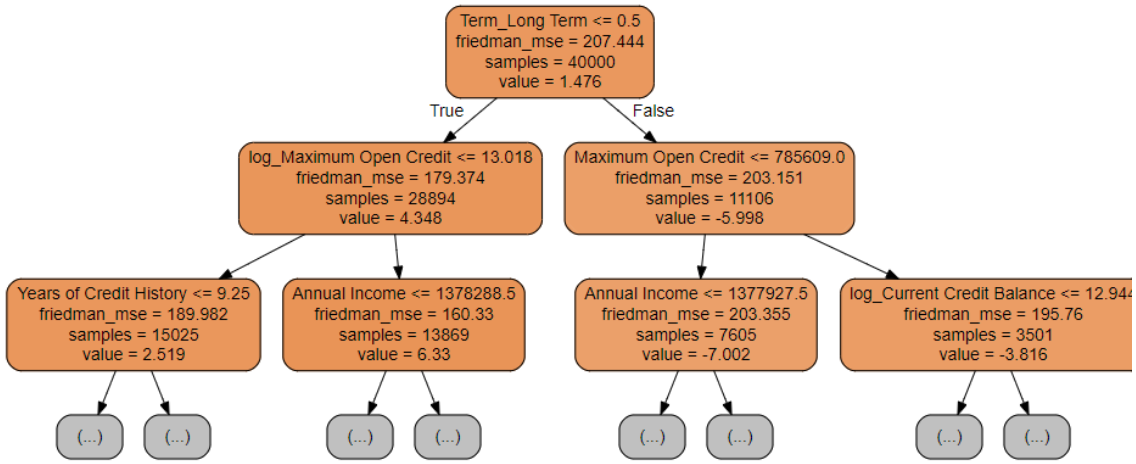


Figure 5.12: Tree Interpretation of GBR

# Chapter 6

# Experimental Results and Analysis

## 6.1 Comparative Analysis of Supervised Models

Initially, we fit our training data to a few well-known existing models used in literature.

- Linear Regression
- Random Forest Regression
- Gradient Boosting Regression
- K-Nearest Neighbors Regression
- Extreme Gradient Boosting Regression

The following performance was observed on the validation set 6.1.

| Model | Mean Absolute Error | Accuracy |
|---|---|---|
| Linear Regression | 18.827 | 88.65% |
| Random Forest Regression | 16.612 | 89.1% |
| Gradient Boosting Regression | 18.363 | 88.93% |
| K-Nearest Neighbors Regression | 19.112 | 88.48% |
| Extreme Gradient Boosting Regression | 18.362 | 88.93% |

Table 6.1: Comparison of Initial models

After conducting Random Search over parameters of each individual model the results improved by a margin as follows.

| Model | Mean Absolute Error | Accuracy |
|---|---|---|
| Random Forest | 13.471 | 91.88% |
| Extreme Gradient Boosting | 15.362 | 90.76% |
| Gradient Boosting Regressor | 15.363 | 90.74% |
| Linear Regression | 16.824 | 89.86% |
| K-Nearest Neighbors | 17.112 | 89.69% |

Table 6.2: Accuracy of different models after initial tuning

Since both time and computational complexity of conducting tuning on all the models was unfeasible. We decided to Perform extensive analysis and hyper-parameter tuning on the top two models with the highest accuracy. Random Forests and Gradient Boosting Regression both based on Regression Decision trees. Randomized Search over the hyper-parameters and 4-fold Cross Validation was undertaken on the Random Forest and GBR models.

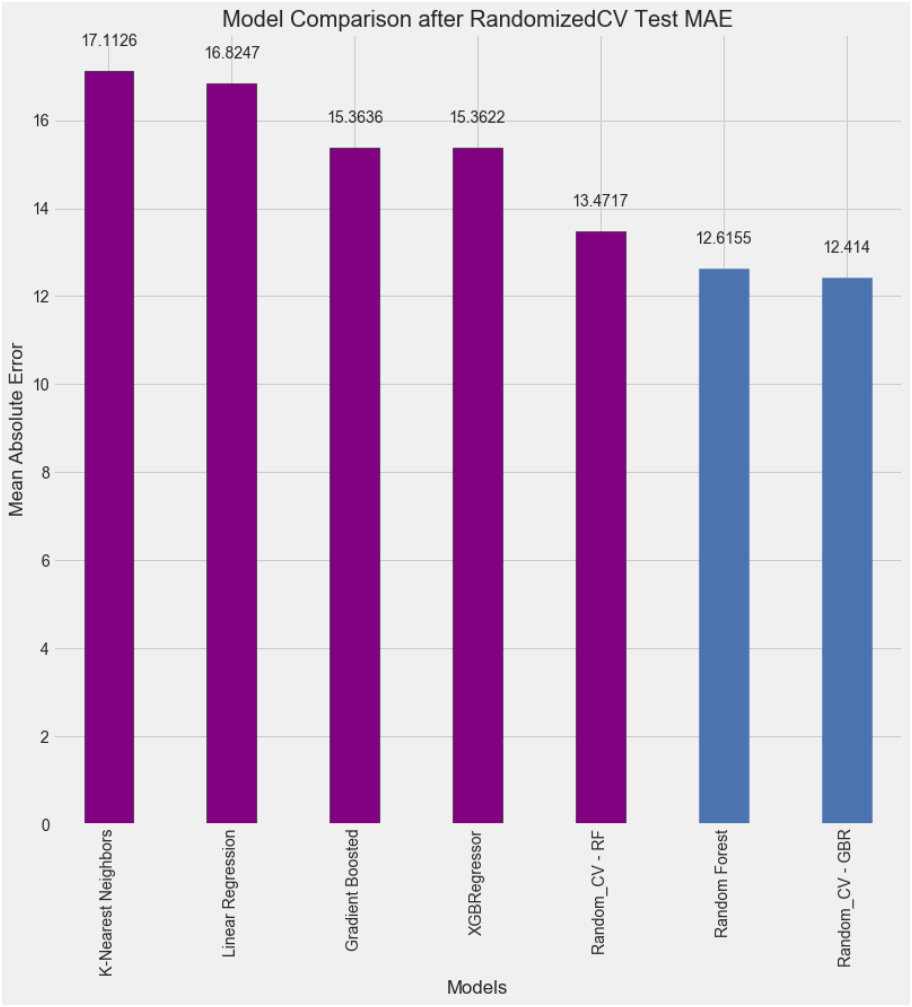


Figure 6.1: Comparison of models after Random Search and Cross-Validation on RF and GBR

Further filtering was done, choosing only the best model(GBR) after Random-CV Search. Grid Search over the hyper-parameters and 5-fold Cross Validation was undertaken, A maximum accuracy of (93%) was achieved on the test set using the optimized GBR model.

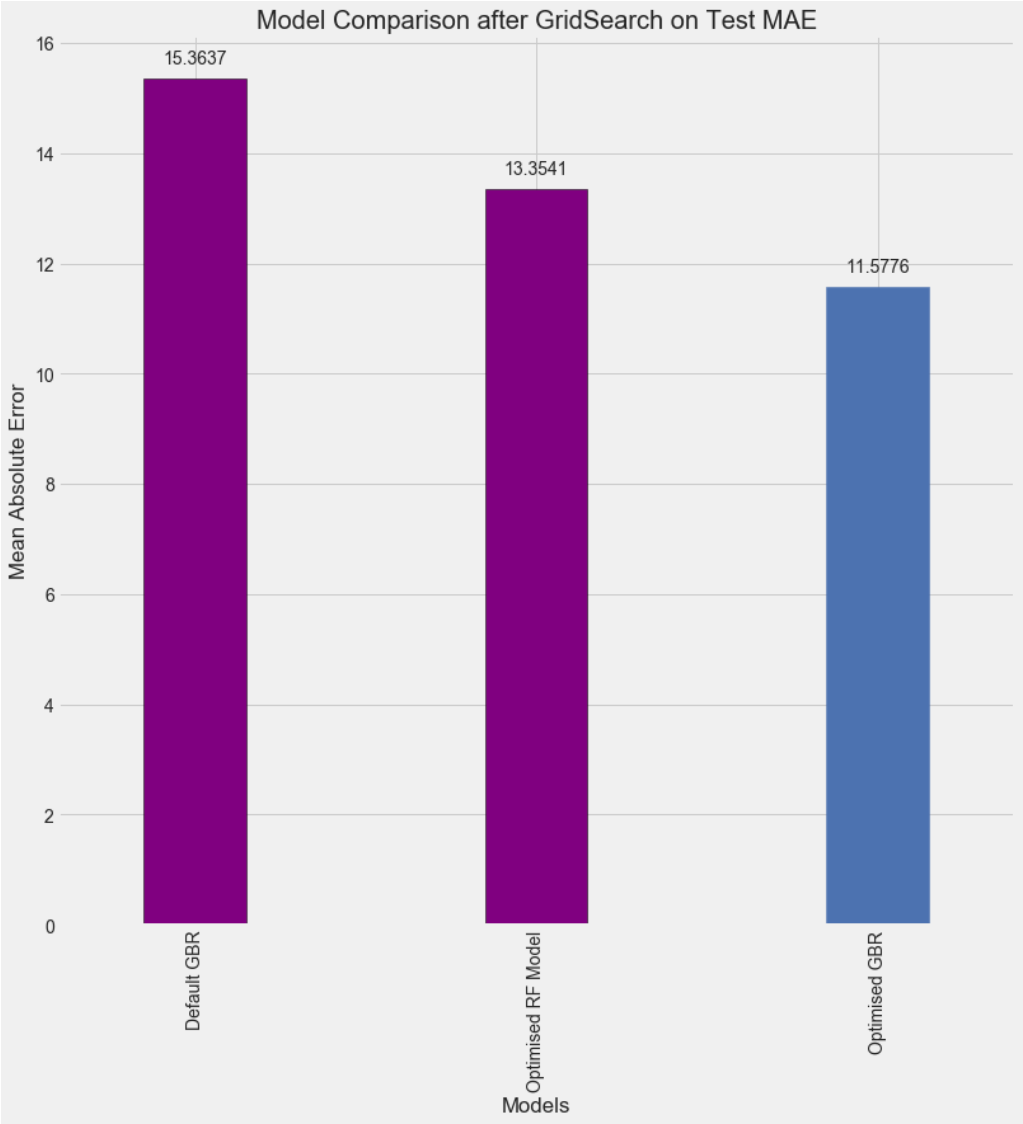


Figure 6.2: Comparison of models after Grid Search and Cross-Validation on GBR

## 6.2 Final Model Evaluation

The final model with the tuned hyper-parameters was fit on the training set. And evaluated using the test set. The model outperformed all the previous models, as well as the default un-tuned Gradient Boosting Regressor Model by a large margin in terms of Mean Absolute error 6.3

| Model | MAE | accuracy |
|---|---|---|
| GridCV - Gradient Boosting Regressor | 11.57 | 93.02% |
| RandomCV - Gradient Boosting Regressor | 12.413 | 92.52% |
| RandomCV - Random Forest | 12.615 | 92.40% |
| Default Random Forest | 13.471 | 91.88% |
| XGBRegressor | 15.362 | 90.76% |
| Default Gradient Boosting Regressor | 15.363 | 90.74% |
| Linear Regression | 16.824 | 89.86% |
| K-Nearest Neighbors | 17.112 | 89.69% |

Table 6.3: Accuracy of all the Models

The predictions generated by the model and the true values had a similar distribution 6.3. The model was good in predicting credit score values below the eligibility threshold (i.e 710). So the model was good in predicting defaulters. It was not so accurate in predicting the bi-modal distribution of credit score above the eligibility threshold. That is not a major concern because the purpose of credit scoring is to accurately predict the defaulters and non-eligible.

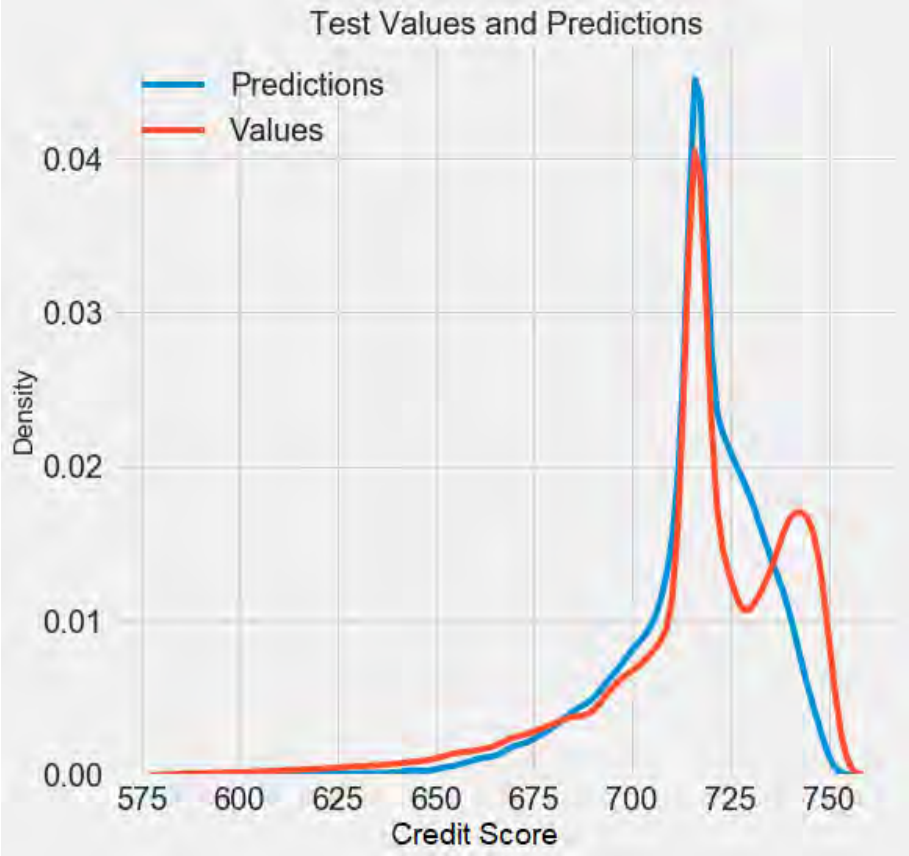


Figure 6.3: KDE plot of Predictions and True Values

Another diagnostic plot is a histogram of the residuals. Ideally, we would hope that the residuals are normally distributed, meaning that the model is wrong the same amount in both directions (high and low) 6.4 [41].

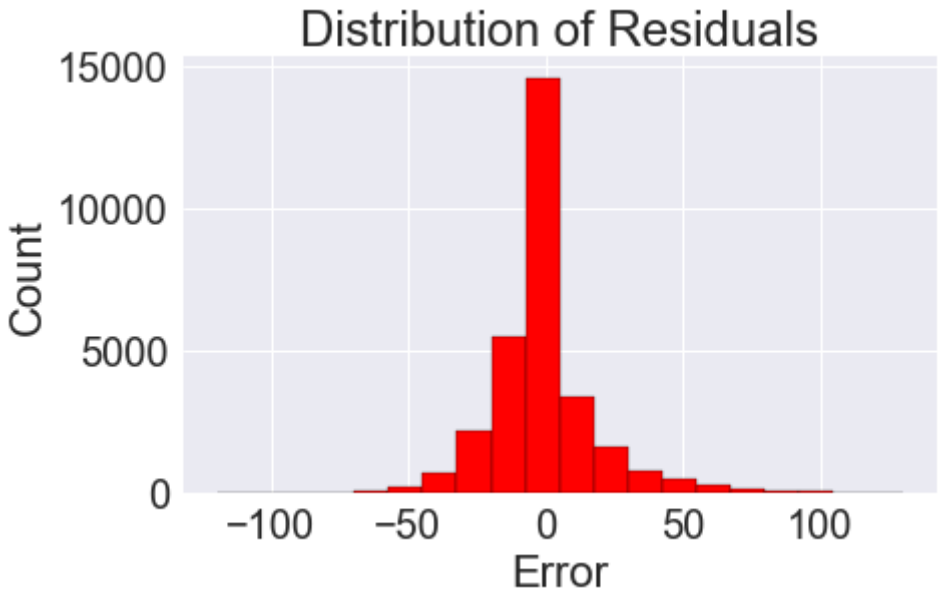


Figure 6.4: Residuals of Error

## 6.3 RfDNN Result Analysis

The RfDNN model could only be conducted using the Random Forest model, since it creates in-dependant base learners or Decision trees which could be used as the input downstream of the DNN architecture.
Experimental analysis was carried out on different values of trees for both the Default Random Forest Regressor as well as the proposed Stacked RfDNN model. It was observed that the proposed model performed better in all the cases, when validated on a test set of 25000 instances.

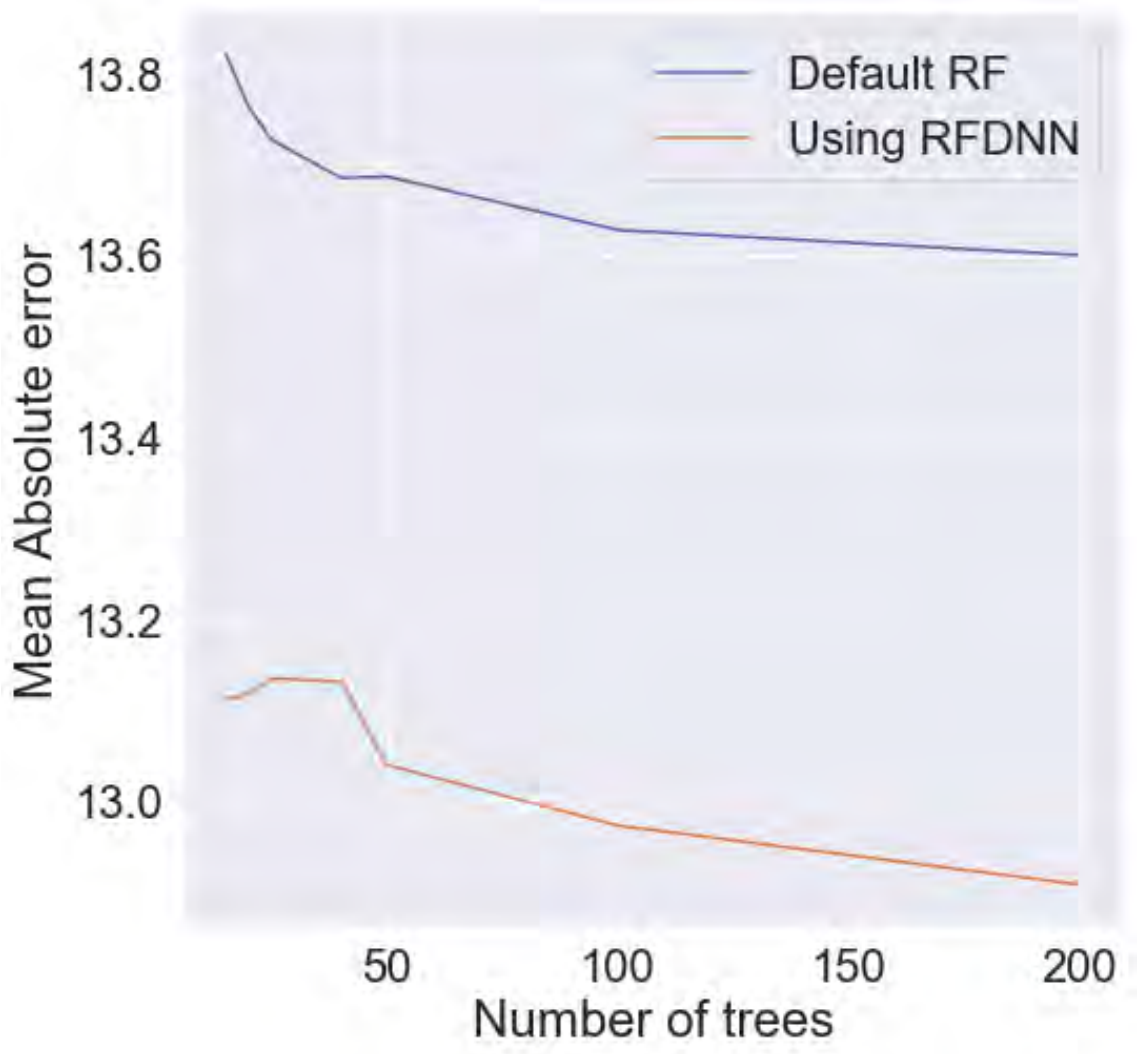


Figure 6.5: Comparison between RF and RF-DNN for different number of trees used

The results of the experiments can be summarized in the table below6.4:

| Trees | RFtrainLoss | RFvalid MAE | RFDNN Valid MAE |
|---|---|---|---|
| 15.0 | 10.518305 | 13.820628 | 13.110735 |
| 20.0 | 10.480548 | 13.761950 | 13.116607 |
| 25.0 | 10.389398 | 13.725964 | 13.133844 |
| 40.0 | 10.369833 | 13.684292 | 13.130353 |
| 100.0 | 10.325133 | 13.627056 | 12.971994 |
| 200.0 | 10.319143 | 13.599303 | 12.907024 |

Table 6.4: RFDNN and RF results

The hyper-parameters of the RfDNN model was chosen in trail and error basis. We could not conduct a proper grid search over all the possible combinations of different hidden layers, activation functions, regularization layers, Batch Norm$(\alpha, \beta)$,learning rate, learning rate annealing and so on.

# Chapter 7

# Conclusion and Future Work

## 7.1 Conclusion and Future Work

To conclude, we were able to achieve our objective of filtering out the best supervised regressor to predict a reference score based on financial and profile data. It was also observed in our experiments that tree-based models perform better in recognizing patterns in tabular data consisting of mostly numeric values. In a debate over which methodology to choose when optimizing a supervised model, we found out that Randomized and Grid Search over the hyper-parameter space along with 4-fold Cross Validation is a reliable option.
Although it should be noted that the time complexity of such an approach is expensive. In our case with mediocre processing power at our dispense, optimizing a model and finding the correct hypothesis space took an average of 6-7 days of continuous training and validating. Trying some other methodologies of narrowing down the hypothesis space by use of neural networks might be a possibility for future work.
At the same time, we would be looking forward to test our model and work-flow on real-world data. This would provide us a greater insight into how to improve our proposal to bring about positive change in the defaulter issue.
Therefore we would like to come to an end with the statement that this paper illustrates an interesting approach in predicting a credit score of an individual. In the current ever changing economy implementation of such a system can bring about remarkable results which in turn can play a major role in assessing credit risk of borrowers and enable all the financial institutions to keep operating in a transparent and profitable way.

# Bibliography


[1] B. L. Lucian, *Bagging, boosting and stacking in machine learning*, Apr. 1962. [Online]. Available: https://stats.stackexchange.com/questions/18891/bagging-boosting-and-stacking-in-machine-learning.

[2] J. R. Quinlan, "Induction of decision trees", *Machine learning*, vol. 1, no. 1, pp. 81–106, 1986.

[3] D. F. Specht, "A general regression neural network", *IEEE transactions on neural networks*, vol. 2, no. 6, pp. 568–576, 1991.

[4] L. J. Mester, "What's the point of credit scoring?", *Federal Reserve Bank of Philadelphia Business Review*, pp. 3–16, Sep. 1997.

[5] D. West, "Neural network credit scoring models", *Computers & Operations Research*, vol. 27, no. 11-12, pp. 1131–1152, 2000.

[6] L. Breiman, "Random forests", *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[7] S. Hochreiter, Y. Bengio, P. Frasconi, J. Schmidhuber, *et al.*, *Gradient flow in recurrent nets: The difficulty of learning long-term dependencies*, 2001.

[8] L. Yu, S. Wang, and K. K. Lai, "Credit risk assessment with a multistage neural network ensemble learning approach", *Expert systems with applications*, vol. 34, no. 2, pp. 1434–1444, 2008.

[9] A. Khashman, "Neural networks for credit risk evaluation: Investigation of different neural models and learning schemes", *Expert Systems with Applications*, vol. 37, no. 9, pp. 6233–6239, 2010.

[10] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines", in *Proceedings of the 27th international conference on machine learning (ICML-10)*, 2010, pp. 807–814.

[11] B. Twala, "Multiple classifier application to credit risk assessment", *Expert Systems with Applications*, vol. 37, no. 4, pp. 3326–3336, 2010.

[12] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, *et al.*, "Scikit-learn: Machine learning in python", *Journal of machine learning research*, vol. 12, no. Oct, pp. 2825–2830, 2011.

[13] C. Vens and F. Costa, "Random forest based feature induction", in *2011 IEEE 11th International Conference on Data Mining*, IEEE, 2011, pp. 744–753.

[14] S. Oreski, D. Oreski, and G. Oreski, "Hybrid system with genetic algorithm and artificial neural networks and its application to retail credit risk assessment", *Expert systems with applications*, vol. 39, no. 16, pp. 12 605–12 617, 2012.

[15] A. Vellido, J. D. Mart in-Guerrero, and P. J. Lisboa, "Making machine learning models interpretable.", vol. 12, pp. 163–172, 2012.

[16] H. A. Bekhet and S. F. K. Eletter, "Credit risk assessment model for jordanian commercial banks: Neural scoring approach", *Review of Development Finance*, vol. 4, no. 1, pp. 20–28, 2014.

[17] A. Gelzinis, A. Verikas, E. Vaiciukynas, M. Bacauskiene, J. Minelga, M. Hallander, V. Uloza, and E. Padervinskis, "Exploring sustained phonation recorded with acoustic and contact microphones to screen for laryngeal disorders", Dec. 2014, pp. 125–132. DOI: 10.1109/CICARE.2014.7007844.

[18] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization", *arXiv preprint arXiv:1412.6980*, 2014.

[19] A. Tang and J. T. Foong, "A qualitative evaluation of random forest feature learning", in *Recent Advances on Soft Computing and Data Mining*, Springer, 2014, pp. 359–368.

[20] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift", *ArXiv*, vol. abs/1502.03167, 2015.

[21] M. A. Islam, "Credit rating operation and effectiveness", *Internship Report Brac University*, Dec. 2015. [Online]. Available: http://dspace.bracu.ac.bd/xmlui/bitstream/handle/10361/4702/13364016.pdf?sequence=1&isAllowed=y.

[22] N. Mohammad and A. N. Onni, "Credit risk grading model and loan performance of commercial banks in bangladesh", *European Journal of Business and Management*, vol. 7, no. 13, pp. 83–91, 2015.

[23] Z. Zhao, S. Xu, B. H. Kang, M. M. J. Kabir, Y. Liu, and R. Wasinger, "Investigation and improvement of multi-layer perceptron neural networks for credit scoring", *Expert Systems with Applications*, vol. 42, no. 7, pp. 3508–3516, 2015.

[24] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[25] Z. C. Lipton, "The mythos of model interpretability", *arXiv preprint arXiv:1606.03490*, 2016.

[26] R. G. Lopes, R. N. Carvalho, M. Ladeira, and R. S. Carvalho, "Predicting recovery of credit operations on a brazilian bank", in *2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA)*, IEEE, 2016, pp. 780–784.

[27] M. T. Ribeiro, S. Singh, and C. Guestrin, "Model-agnostic interpretability of machine learning", *arXiv preprint arXiv:1606.05386*, 2016.

[28] ——, "Why should i trust you?: Explaining the predictions of any classifier", pp. 1135–1144, 2016.

[29] G. V. Attigeri, M. Pai, and R. M. Pai, "Credit risk assessment using machine learning algorithms", *Advanced Science Letters*, vol. 23, no. 4, pp. 3649–3653, 2017.

[30] "Bad loans cripple the banking sector", *Dhaka Tribune*, Oct. 2017.

[31] Z. Begiev, *Bank loan status dataset*, Apr. 2017. [Online]. Available: https://www.kaggle.com/zaurbegiev/my-dataset#credit_train.csv.

[32] A. Lawi, F. Aziz, and S. Syarif, "Ensemble gradientboost for increasing classification accuracy of credit scoring", in *2017 4th International Conference on Computer Applications and Information Processing Technology (CAIPT)*, Aug. 2017, pp. 1–4. DOI: 10.1109/CAIPT.2017.8320700.

[33] *What is the difference between bagging and boosting?*, Oct. 2017. [Online]. Available: https://quantdare.com/what-is-the-difference-between-bagging-and-boosting/.

[34] P. Addo, D. Guegan, and B. Hassani, "Credit risk analysis using machine and deep learning models", *Risks*, vol. 6, no. 2, p. 38, 2018.

[35] AnnieReporter, "More than 1 million people default on their student loans each year", *CNBC*, Aug. 2018. [Online]. Available: https://www.cnbc.com/2018/08/13/twenty-two-percent-of-student-loan-borrowers-fall-into-default.html.

[36] P. Ferrando, *Understanding how lime explains predictions*, Dec. 2018. [Online]. Available: https://towardsdatascience.com/understanding-how-lime-explains-predictions-d404e5d1829c.

[37] A. Fisher, C. Rudin, and F. Dominici, "Model class reliance: Variable importance measures for any machine learning model class, from the "rashomon" perspective", *arXiv preprint arXiv:1801.01489*, 2018.

[38] A. Fuster, P. Goldsmith-Pinkham, T. Ramadorai, and A. Walther, "Predictably unequal? the effects of machine learning on credit markets", *The Effects of Machine Learning on Credit Markets (November 6, 2018)*, 2018.

[39] P. T. of India, "9 million loan defaulters blacklisted in china $27 billion frozen", Jan. 2018.

[40] Y. Kong and T. Yu, "A deep neural network model using random forest to extract feature representation for gene expression data classification", *Scientific Reports*, vol. 8, no. 1, p. 16 477, 2018, ISSN: 2045-2322. DOI: 10.1038/s41598-018-34833-6. [Online]. Available: https://doi.org/10.1038/s41598-018-34833-6.

[41] H. Ma, X. Yang, J. Mao, and H. Zheng, "The energy efficiency prediction method based on gradient boosting regression tree", in *2018 2nd IEEE Conference on Energy Internet and Energy System Integration (EI2)*, IEEE, 2018, pp. 1–9.

[42] G. Mowla, "Default loans plague banking sector", 2018.

[43] J. Nalić and A. Švraka, "Importance of data pre-processing in credit scoring models based on data mining approaches", in *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, May 2018, pp. 1046–1051. DOI: 10.23919/MIPRO.2018.8400191.

[44] J. Nalić and A. Švraka, "Using data mining approaches to build credit scoring model: Case study—implementation of credit scoring model in microfinance institution", in *INFOTEH-JAHORINA (INFOTEH), 2018 17th International Symposium*, IEEE, 2018, pp. 1–5.

[45] A. Nova, *More than 1 million people default on their student loans each year*, Aug. 2018. [Online]. Available: https://www.cnbc.com/2018/08/13/twenty-two-percent-of-student-loan-borrowers-fall-into-default.html.

[46] P. Grover, *Gradient boosting from scratch*, Aug. 2019. [Online]. Available: https://medium.com/mlreview/gradient-boosting-from-scratch-1e317ae4587d.

[47] C. Molnar, *Interpretable machine learning*, Sep. 2019. [Online]. Available: https://christophm.github.io/interpretable-ml-book/feature-importance.html.

[48] I. Staff, "The importance of your credit rating", *Investopedia*, Jun. 2019. [Online]. Available: https://www.investopedia.com/articles/00/091800.asp.

[49] Tonester524, *Understanding random forest*, Aug. 2019. [Online]. Available: https://towardsdatascience.com/understanding-random-forest-58381e0602d2.

[50] Vishalmorde, "Xgboost algorithm: Long may she reign!", *Medium*, Apr. 2019. [Online]. Available: https://towardsdatascience.com/https-medium-com-vishalmorde-xgboost-algorithm-long-she-may-rein-edd9f99be63d.

[51] L. Weston, "Why your credit score is important", *NerdWallet*, Mar. 2019. [Online]. Available: https://www.nerdwallet.com/blog/finance/great-credit-powerful-tool/.

[52] [Online]. Available: http://www.assignment.com/.

[53] "Decision tree regression", *Decision Tree Regression*, [Online]. Available: https://www.saedsayad.com/decision_tree_reg.html.