**US Corporate Bankruptcy Data Analysis & Prediction using Machine Learning Algorithm, Python**



THESIS SUBMITTED TO

Symbiosis Institute of Geoinformatics

FOR PARTIAL FULFILLMENT OF THE M. Sc.(DATA SCIENCE AND SPATIAL ANALYTICS) DEGREE

By

Shuvam Sanyal

(Batch: -2019-2021)

Symbiosis Institute of Geoinformatics

Symbiosis International (Deemed University) 5th Floor, Atur Centre, Gokhale Cross Road

Model Colony, Pune - 411016

© 2020 Symbiosis International (Deemed University), Pune, India

# Certificate

Certified that the thesis titled “**US Corporate Bankruptcy Data Analysis & Prediction using Machine Learning Algorithm, Python (1979-2017)**” is a bonafied work done by **Mr. Shuvam Sanyal** at Symbiosis Institute of Geoinformatics, under our supervision.

# Supervisor, Internal

**(Name): - Prof Dr. Vidya Patkar**

**Organization Name: - Symbiosis Institute of Geoinformatics**

**Acknowledgments**

This thesis was written during the summer of 2020, as part of my Master of Science degree summer project in Data Science and Spatial Analytics, majoring in Data Science, with a specialization in Machine Learning, Deep Learning, Spatial Analytics and Artificial Intelligence.

I have found the study to be challenging as well as rewarding, especially taking into consideration the vast amount of data. I strongly believe that my ﬁndings will help in improving bankruptcy predictions in US or around the world. I also believe, compared to the amounts of work have been done in this domain, my work has some new interesting applications of Machine Learning for Bankruptcy Prediction.

I would like to express my deepest gratitude to our supervisor and mentor. Dr. Vidya Patkar, the assistant professor of Symbiosis Institute of Geoinformatics under Symbiosis International (Deemed University), Pune, India. She read drafts, gave advice, and saw the project through from inception to completion. With her support, input and guidance, the quality and actuality of my research have greatly improved. Additionally, I would like to thank our director Dr, T.P. Singh, professors Ms. Darshana Pathak and Dr. Navendu Chowdhury for helping us with the project and internal guide allocation. I would also like to thank the Data Science experts on LinkedIn who helped me solving my doubts and queries during my project work. Last but not the least, thanks to all data science fraternities and everyone associated with Symbiosis Institute of Geoinformatics for giving me a chance to be a part of the exciting, rewarding and challenging project.

# Project Index

|  |  |
| --- | --- |
| **One| Abstract of the project** | 4 |
| **Two | List of Figure** | 5 |
| The List of Figures Used in This Project | 5 |
| **Three | List of Tables** | 6 |
| List of Tables Used in This Project | 6 |
| **Four |Content** | 7-9 |
| **Five |Abbreviation List** | 10 |
| **Six | Preface** | 11 |
| **Seven | Introduction to the Project** | 12 |
| Economics Behind Bankruptcy | 12 |
| An Example of Real-World Bankruptcy Case | 12 |
| **Eight | Literature Review** | 13-14 |
| **Nine| Data-set Discussion (Study Area)** | 15 |
| **Ten| Methodology** | 16-26 |
| **Eleven| Result** | 26-37 |
| **Thirteen| Discussion** | 38-39 |
| **Fourteen|Conclusion** | 40 |
| **Fifteen |References** | 41 |
| **Sixteen|Annexure and Appendix** | 42-47 |

**Abstract of the project**

Since Lehman Brother bankruptcy catastrophe event during 2008 global financial crisis, estimating the advanced risk of corporate bankruptcies has been of large importance to creditors and investors. Despite being a relatively new research topic, in recent years, artificial intelligence and machine learning methods have achieved promising results in corporate bankruptcy prediction settings. In this research summer project, I created a new interesting machine learning model for predicting upcoming bankruptcies using around 46 years US Corporate Bankruptcy Dataset. After thorough cleaning and missing value imputation as well as feature engineering, our ﬁnal dataset finally contains 23320 observations with 210 features related to ﬁnancial, management statements from 93837 observations with 15 features.

I performed my analysis based on nine different machine learning techniques (Logistic Regression, KNN, SVM, Naïve Bayes, Decision Tree, Random Forest, AdaBoost, XgBoost, CatBoost) on the dataset. For evaluation, I have used Accuracy Scores, ROC-AUC Curve, Confusion Matrix, Precision, Recall as well as F Score and Cumulative Gain Chart. The best models came out to be Random Forest, XgBoost, AdaBoost, CatBoost and Decision Trees. After applying an Ensemble Voting Method on top 5 algorithms, it votes for Random Forest and the boosting algorithms to be the best two predictors on both training and testing data of bankruptcy cases, thus reducing over fitting problem. To crosscheck over fitting, we used the cross-validation method to find CV mean scores of algorithms. Again, Random Forest & Gradient Boosting algorithms topped the list. Both of the algorithms yields an overall accuracy of ∼93%, training data accuracy of ∼99% and class independent test accuracy of∼92% on the balanced imputed and feature engineered dataset, which I over sampled using SMOTE along with dimension reductions using PCA even before performing the Machine Learning models to achieve better accuracies. My final model is finally able to correctly predict all 8033 bankrupt ﬁrms correctly and 17208 non-bankrupt ﬁrms correctly out of 17217. Only 9 corporations out of all, have been misclassified as bankrupt when they are actually not. The results, I obtained from KNN, SVM, Naïve Bayes and Logistic Regression take lots of time to train and do not perform good on test datasets as well as training the model, despite having few good accuracies, when dataset is really large. Hence these models are not recommended for any kind of Corporate Bankruptcy or financial predictions. Furthermore, I found that our model assigns importance to few of the individual components of their ratios, in particular, components related to asset, liquidity, profitability and productivity.

Keywords– Bankruptcy Prediction, Machine Learning, Logistic Regression, KNN, SVM, Naïve Bayes, Decision Tree, Random Forest, AdaBoost, XgBoost, Ensembled based voting method and Cat Boost, ROC-AUC Curve, Confusion Matrix, Precision, Recall, F Score, SMOTE, Cumulative Lyft and Gain Chart.

|  |  |
| --- | --- |
| **List of Figures** |  |
| 2.1 Pie Chart for Proportion of Bankruptcy in Dataset……………………………………… | 28 |
| 2.2 Data Distribution Using Density Subplots………………………………………………. | 29 |
| 2.3 Year wise plot for Bankruptcy Over the Years………………………………………... | 27 |
| 2.4 Nullity Matrix for Missing Data Patterns Visualization………………………………… | 26 |
| 2.5 Missing Data Bar Plot Representation …………………………………………………. | 27 |
| 2.6 Missing Value Nullity Correlation Analysis Using Heat Map…………………………. | 26 |

* 1. Heatmap for Checking Correlation Among All Variables 26
  2. Regression Missing Value Imputation Visualization 20
  3. Stochastic Regression Missing Value Imputation Visualization 20
  4. Box Plot Visualization 29
  5. Over Sampling of Imbalanced Data Performance 22
  6. KNN Accuracy Performance and Confusion Matrix 30
  7. Logistic Regression Accuracy Performance and Confusion Matrix 30
  8. SVM Accuracy Performance 34
  9. Decision Tree Accuracy Performance and Confusion Matrix 31
  10. Random Forest Accuracy Performance and Confusion Matrix 31
  11. Ada Boost Accuracy Performance and Confusion Matrix 32
  12. Xg Boost Accuracy Performance and Confusion Matrix 32
  13. Cat Boost & Naïve Bayes Accuracy Performance 33
  14. Ensemble Voting Classifier Accuracy Performance and Confusion Matrix 34
  15. Cross Validation Performance of Different Top Classifiers 35
  16. ROC Curves for all Machine Learning Classifiers 35
  17. Overall Confusion Matrix, Precision, Recall, F Score Figure 36-37
  18. Figure of Variable Importance 46

2.25 Cumulative Gain and Lyft Chart 46

# List of Tables

3.1Confusion Matrix Tables for all Classifier 36

|  |  |
| --- | --- |
| 3.2 F1 Score Table for all Classifier………………………………………………………. | 30-34 |
| 3.3 Accuracy Table for All Classifiers……………………………………………………. | 30-34 |
| 3.4 Over Sampling Table of Balanced Target Class……………………………………… | 22 |
| 3.5 Accuracy, F1, Recall, Precision Tables for All Classifier……………………………. | 37 |
| 3.6 Principal Component Table…………………………………………………………... | 21 |
| 3.7 Feature Importance Table Based on Best Classifier…………………………………. | 46 |

# Content

1. **Introduction**
   1. Economics Behind Bankruptcy 12
   2. An Example of Real-World Bankruptcy Case 12

# Literature Review

* 1. Standard Models 13-14
     1. Early Adaptions……………………………………………………………13-14
     2. Altman Z-Scores 13-14
     3. Ohlson O-Score 13-14
     4. Random Forest and Logistic Regression 13-14
  2. Specialized Models 13-14
     1. Quarterly Predictions 13-14
     2. Bloomberg’s Bankruptcy Predictor 13-14
  3. Recently used Models 13-14

# Study Area

* 1. Dataset Description 15
  2. Input and Output Variables Explanation 15

# Methodology

* 1. Importing all necessary libraries and packages 17
  2. Importing and organizing the data in csv format 18
  3. Explanatory Data Analysis and Data Pre-processing 18
     1. Missing Data Analysis 18
     2. Different Data Imputation Techniques and Selecting the Best One 19-20
     3. Generate few new features using feature matrix 21
  4. Machine Learning Modelling & Prediction……………………………………21-24 7
     1. Test Train Split 21
     2. Dimensionality Reduction Using PCA, LDA, QDA ……………………21
     3. Oversampling and Imbalance Target Class Handling 22
     4. K Nearest Neighbours 22
     5. Logistic Regression 23
     6. Decision Tree 23
     7. Random Forest 23
     8. Support Vectors Machine 23
     9. AdaBoost 23
     10. Xg boost 24
     11. Naïve Bayes 24
     12. Cat boost 24
     13. Ensemble Method using Voting… 24
  5. Evaluation of Model and Validation 25-26

4.5.1 Cross-Validation 25

4.5.2Precision, Recall, F Score 25

* + 1. ROC AUC Curve 25
    2. Confusion Matrix 26
    3. Cumulative Lyft and Gain Chart 26

# Results

5.1 Base case Threshold 26-37

5.1.1 Missing Value and Data Pre-Processing Visualisation……………………26-29

* + 1. K Nearest Neighbours 30

5.1.3 Logistic Regression 30

* + 1. Decision Tree 31
    2. Random Forest 31
    3. Xg boost 32
    4. Ada boost 32
    5. Naïve Bayes 33
    6. Cat Boost 33
    7. Ensemble Method… 34
    8. Support Vectors Machine 34
    9. Comparison 35
    10. Cross Validation Mean Score Result 35
    11. ROC AUC Curve Result 35
    12. Confusion Matrix Result 36
    13. Overall Precision, Recall and F Score 37

# Discussion

* 1. Comparison to the Literature 38

8

* 1. Discussion of Dimensionality 38
  2. Comparison of Variable Selection 39
  3. Discussion on Applications 39
  4. Further Research 39

1. **Conclusion** 40

# Reference 41

# Appendix and Annexure 42-47

A1 Technical Notes on Bankruptcy 42

A2 Correlation Matrix 42

A3 Automated Feature Engineering… 43

A4 Mathematics behind Top Performing Machine Learning Algorithms 43-45

A5 Accuracy table for all classifiers 45

A6 Cumulative Gain Chart 46

A7 Variable Importance 46

A8 Computer Speciﬁcations 47

**Abbreviation list**

* **KNN: -** K Nearest Neighbor Classifier
* **SVM: -** Support Vector Machine Classifier
* **XgBoost: -** Xtreme Gradient Boosting Classifier
* **AdaBoost: -** Adaptive Gradient Boosting Classifier
* **Cat Boost: -** Categorical Gradient Boosting Classifier
* **EPS: -** Earning Per Share
* **BK: -** Bankruptcy Status
* **Market Book Ratio: -** Market Rate of Company’s Assets/Book (Balance Sheet) Rate of Company’s Asset
* **PCA**: - Principal Component Analysis
* **MICE Imputation**: - Multiple Imputation Chained Equation
* **KNN Imputation**: - K Nearest Neighbor Imputation
* **ROC Curve**: - Receiver Operating Characteristic Curve
* **AUC: -** Area Under Curve
* **TPR: -** Total Positive Rate
* **FPR: -** False Positive Rate
* **FNR: -** False Negative Rate
* **TNR: -** Total Negative Rate
* **SMOTE**: - Synthetic Minority Over-sampling Technique.
* **ADASYN**: -Adaptive Sampling Method for Imbalanced Data.

# Preface

This research summer project report is written as a part of my Master of Science (MSc) degree in Data Science at the Symbiosis Institute of Geoinformatics under Symbiosis International (Deemed University), Pune, India, summer 2020. I am majoring in Data Science, Machine Learning, Deep Learning, Spatial Analytics. So, it was therefore important to find a topic that would cover my education programs. I also wanted to put my previous undergraduate and post bachelor work experience with statistics to use and that is how I ended up using a dataset on bankruptcy data.

The project is based on a dataset of US Corporate Bankruptcy figures with 92873 rows of financial feature variable datapoints from year 1979 up to 2017. The column “BK” in the dataset provided denotes whether the company is bankrupt (indicated by 1/Yes) or not (indicated by 0/No).

The particular focus on bankruptcies was somewhat inspired by the recent Corona Virus pandemic and its fierce adverse effect on the world economy and corporations.

I would sincerely like to thank the people listed above in acknowledgement section. Most importantly, I would like to thank my supervisor Assistant Professor Dr Vidya Patkar again for the help and support I have received during this semester. She has been available to let me through the entire process and her guidance and feedback have been incredibly valuable.

I want to thank my LinkedIn experts, Data Science professionals who have contributed with comments and suggestions to my project, and also for being supportive and patient with me all along.

# Introduction of the project

**1.1** **Economics Behind Bankruptcy**: -

Corporate bankruptcy is some kind of legal way of handling those businesses which are unable to repay their outstanding debts. At first the assests of all the debtors gets calculated and then if possible, a part of those assets can be used as a source of repayment.

The organization’s solvency is calculated using the probability value of the scenario where the company is not able to pay its debt. The main frameworks of corporate bankruptcy are: - (1) the smaller the probability of failure, larger the reservoir and net liquid asset operation flow. (2) the greater the probability of failure, greater the amount of debt held and operations fund expanses.

Business insolvency can occur two ways:

* **Receivership**: - A receivership occurs when a bank elects a receiver to sell the assets with security.
* **Liquidation**: - If a business isn’t a potentially salvageable, the business is wound up by either application for voluntary liquidation or liquidation processes conducted by a liquidator appointed either by creditors or a court.

Few financial ratios that influence business insolvencies are: - 1) Profitability 2) Costs 3) Capital turnover 4) Liquidity 5) Asset structure 6) Capital structure 7) Growth 8) Liquidity 9) Tobin’s Q.

**1.2 An Example of Real-World Bankruptcy Case: -**

Back in 2008, Lehman Brothers Holdings, Inc. filed for the biggest bankruptcy event in the history of United States with $639 billion in assets and $613 billion in debts. At that particular time, it was the fourth-largest U.S. investment bank. The cause of Lehman’s fall was its large exposure to the

U.S. subprime mortgage and real estate markets. Following the trend of most of the IB banks, Lehman Brothers was banking on mostly short-term markets to raise billions of dollars each day. Ultimately, it was an inability to secure funding that causes this sad ending for them.

Since Lehman Brother bankruptcy catastrophe event during 2008 global financial crisis, estimating the risk of corporate bankruptcies has been of large importance to creditors and investors. Currently due to this ongoing global coronavirus pandemic, all leading experts predicted that the economic depression, we are going to face in coming few years, will be much higher on scale than what we faced in 2008. Few of the top firms which have either filed for bankruptcy of about to file around the world are Virgin Australia airline, Rubie’s Costume Company, Swedish fashion retail chain MQ, J. Crew, Frontier Communications], British airline Fly be & Virgin airline, to name a few. So, prior prediction of bankruptcy automatically becomes a natural choice to save corporations and economy as a whole in these kinds of economic emergencies. Despite being a relatively new research topic, in recent years, artificial intelligence and machine learning based methods have been achieving promising results in the case of corporate bankruptcy prediction. Therefore, in this summer project, various machine learning algorithms I will be using to analyze current scenario and predict future corporate bankruptcy chances in advance based on our dataset using python.

# Literature Review

The amount of research on Bankruptcy Prediction has increased rapidly post 2008 global financial crisis.

# 2.1 Standard Models: -

**2.1.1 Early Adaption: -** FitzPatrick, Paul Joseph. (1932) first time predicted bankruptcy. In his article, he presented the data for 19 pairs of firms and then as part of prediction he compared 13 financial ratios to predict these ratios as probable indicators of failures.But unfortunately, he could not show a meaningful relationship with bankruptcy.

* + **2.1.2 Altman Z-Scores: -** Altman’s Z-Scores (Edward I. Altman, September 1968) is one of the most famous bankruptcy prediction models where he developed Z score model based on some pre-determined financial ratios. He selected two groups with 33 bankrupt and 33 non bankrupct firms and then derived which of the linear combination of the characteristics best way discriminates between these two groups (Altman, 1968 p. 592). He applied Multivariate Discriminant Analysis while compiling 22 potential and important financial ratios for model evaluation. Now out of 22, he selected top 5 ratios based on performance. Companies with a cut off Z score of above 2.67 were classified as Non- Bankrupt. ("Financial Ratios, Discriminant Analysis and the Prediction of Corporate Bankruptcy", Journal of Finance, Vol. 23, No. 4, pp. 589-609).
  + **2.1.3 Ohlson O-Score: -** Ohlson created another famous model back in 1980(James Ohlson, 1980). The model was created by a sample of 105 firms that excluded firms in sectors with different financial structures. From the sample, 17% was listed as bankrupt while the remaining part was non- bankrupt.
  + **2.1.4 Random Forest & Logistic Regression**: - Logistic Regression and Random Forest are the models that have been used extensively for bankruptcy prediction since 1968.However, recently due to emergence of artificial intelligence, new innovative methods like boosting and bagging machine learning models are being applied.

# 2.2 Specialized Models: -

* + **2.2.1 Quarterly Predictions: -** Baldwin and Glezen (J.Baldwin and G.W.Glezen, June 1992,) published a paper where they used a Linear Discriminant Analysis (LDA) by using 24 variables on quarterly data. They ended up predicting bankruptcy more accurately seven quarter prior to the actual event which was surprisingly better than previous methods applied.( Journal of Accounting, Auditing & Finance;Summer92, Vol. 7 Issue 3, p269)
  + **2.2.2 Bloomberg’s Bankruptcy Predictor: -** Bloomberg’s Bankruptcy Predictor DRSK(Cai and Singenellore, 2012) models are actually more into credit modelling. The assumptions made for this were continuous treading, short selling, frictionless training and Brownian motion distribution. The model for non-financial private companies fetched an accuracy of 85.6% to 87.8%.

# 2.3 Recently used and to be explored Models: -

* + Violation of the multivariate discriminant approach since end of 1970 has led researchers to focus their efforts on the development conditional probability model with greater importance on logistic and binary regression. The first application of usage of Neural Network in the prediction of business insolvency was ever published, was the one by Bell,Ribar and Verchio("Neural Nets Versus Logistic Regression: A Comparison of Each Model's Ability to Predict Commercial Bank Failures, Proceedings of the 1990 Deloitte & Touche/University of Kansas Symposium on Auditing Problems 1990, pp.29-53).
  + In recent years, due to the availability of large amount of data, large processing capacity and advanced analysis techniques, Neural Network is the most used algorithm along with Random Forest in modern corporate bankruptcy prediction problem. Also, there are other machine learning methods: Gradient Boosting (XgBoost, AdaBoost, CatBoost) which are people using increasingly to achieve greater accuracies in smarter ways. The results comparison using traditional models like Discriminant Analysis or Logistic Regression and recent algorithms like: Support Vector Machines, Ensemble Methods, Bagging, Boosting Naïve Bayes actually shows there is a huge opportunity to work on new findings in the area of business insolvency prediction.

# Study Area

**3.1 Dataset Description**: -

The data is collected from authorized portal and my main objective is to design and develop machine learning models for future corporate bankruptcy prediction and to show that how modern-day boosting algorithms can bring great accuracies in smarter ways than the traditional algorithms using this data in Python. The dataset contains 92873 rows of financial feature variable datapoints from year 1979 up to 2017 for US based organizations. The column “BK” in the dataset provided denotes whether the company is bankrupt (indicated by 1/Yes) or not (indicated by 0/No).

**3.2 Input and Output Variables Explanation**: -

Initial Input and output variables are described below:

* + - EPS – Earnings Per Share
    - Liquidity – Working Capital/Total Assets
    - Profitability – Retained Earnings/Total Assets
    - Productivity – EBIT/Total Assets
    - Leverage Ratio – (Total Long-term debt + Debt in Current liabilities)/Stockholders
    - Equity Asset Turnover – Sales/ Total Assets
    - Operational Margin – EBIT/Sales
    - Market Book Ratio – (Price Close Annual Fiscal \* Common Shares Outstanding)/Book Value Per Share (This is the company’s current stock price for all outstanding shares relative to its book (balance sheet) value.
    - Asset Growth – Change in assets from previous year
    - Sales Growth – Change in sales from previous year
    - Employee Growth – Change in employees from previous year
    - Tobin’s Q – (Total market value of company + liabilities)/ (Total asset or book value + liabilities)
    - BK – Company bankrupt or not.

# Methodology

Data Collection

Data Modelling

Data Pre processing

Importing all necessary libraries

Test Train Dataset Split

Model Validation and Performance Metrics

Cross Validation and Accuracy Measurement

Dimensionality Reduction

Handling Imbalanced Target Class Dataset

Bankruptcy Prediction (Yes/No)

Explanatory Data Analysis and Data Pre-processing

Applying Machine Learning Algorithms

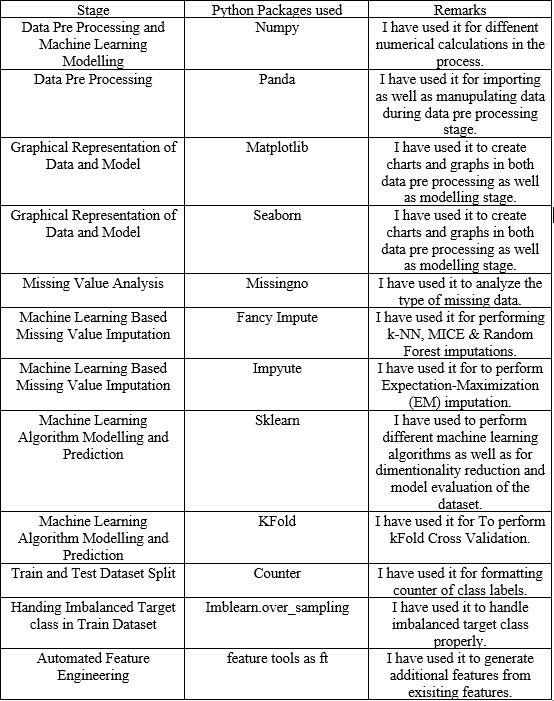
Feature Engineering

Missing Data Imputation using different techniques

Missing Data Analysis

Importing and organizing the data

# 4.1 Importing all necessary libraries and packages: -



# 4.2 Importing and organizing the data in csv format: -

Then I uploaded the dataset on google drive and there after loaded on my google colab python notebook as a panda data frame. The data frame comes with around 92860 rows and 15 columns of datapoints which includes target label of bankruptcy (BK Column) too.

# 4.3 Explanatory Data Analysis and Data Pre-processing: -

Firstly, I have cleaned the dataset by removing duplicates and done feature scaling of the dataset using standardization since the data was highly right screwed and standardization does not get affected by outlier and the distribution of the data is not completely multivariate gaussian. Then I identified the outliers and omitted those.

# 4.3.1 Missing Data Analysis: -

* **Nullity Matrix**: - The nullity matrix helps to quickly visually pick out any kind of missing data patterns in the dataset. Also, the sparkline on the right of the graph gives a summary of the general shape of the data completeness and acts as an indicator of the rows with maximum and minimum rows.
* **Nullity Correlation**: - Nullity correlation heatmap varies from -1 to 1 (-1 ≤ R ≤ 1). Features with zero missing value are excluded in the heatmap. Even if the nullity correlation is very close to zero (-0.05 < R < 0.05), no value will be displayed. In addition to that, a perfect positive nullity correlation (R=1) score shows when the first second feature will both be having corresponding missing values. On the other hand, a perfect negative nullity correlation (R=-1) indicates one of the features to be missing and the second is not missing. Despite having few correlation excesses of 0.5, I have not deleted those variables as they have very important shares in predicting bankruptcies.

# 4.3.2 Different Data Imputation Techniques and Selecting the Best One: -

* **Mean, Median, Mode Imputation**: - This kind of imputation works only when the

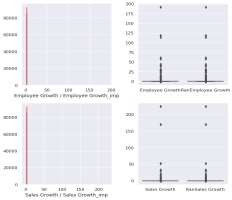
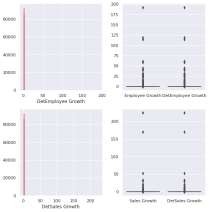
Missing value percentage in the dataset is around 3%, but since the missing value percentage is much higher in my dataset, so I have ignored these basic missing value imputation techniques to achieve better results.

Since missing values in my dataset comes under item non response categories where a particular cell or column data is missing and follows Missing at Random condition, where the missing data depends only on the other features’ observed values, we have selected some of the advanced machine learning prediction model based multiple iterative missing value imputation techniques here to see which one to be selected as the final model for missing value imputation in our dataset in order to achieve best result.

* **KNN Imputation Technique: -** K Nearest Neighbor is a simple classification algorithm which can be very useful in case of missing value imputation. It actually finds feature similarities in K nearest neighbors for the observation with missing values. Then these observations with missing values get replaced by non missing values in the training dataset of these K nearest neighbor. Although it imputes the most similar points but this process can take much time as well as it deletes a lot of datapoints from dataset due to its over sensitiveness to outliers.
* **Multiple Imputation Chained Equation (MICE): -** This MICE Imputation technique of missing value imputation uses multiple iterative stepwise regression models to predict each observation with missing values based on the other observations in the dataset. It initially imputes all the missing observations using mean and then it takes one variable at a time and sets it back to missing. Then it predicts this missing value as a dependent variable in a regression equation using rest of the variables as predicators. This iterative process continues until the whole process converges to optimization. It works perfects with missing values missing at random, introduces statistical uncertainty to lower both bias and variance in the model.
* **Regression and Stochastic Regression Imputation: -** The regression imputation is a kind of missing value imputation techniques where the missing value of each observation in a feature is predicted/estimated based on the values of observations in other variables and then impute those estimated values in cases where values are missing. In the deterministic regression mean value imputation often imputes missing values with exact prediction estimations and thus

18

ignores the random variation in the model. Imputed value therefore leads to an overestimation. Of correlation between X & Y. To bring back the random variation in the data we use stochastic regression imputation by adding a small extra random error with mean zero and variance is equal to the standard error of the regression estimates at the end of the predicted value and thus it brings back the random variation in the dataset. Now even stochastic regression imputation may lead to implausible values where the variable is often too much restricted to certain range of values (like age should be positive always) introducing bias sometimes just like deterministic regression imputation which also does introduce bias. Apart from that stochastic regression imputation technique do not perform accurately when there is heteroscedasticity in data. So, sometimes we look for better alternatives of these imputation techniques.



Regression Imputation Visualization Stochastic Regression Imputation Visualization

From the above visualization we can see both of these imputation techniques are somehow prone to skewness of the data and introducing bias. So better techniques than simple linear/stochastic regression imputation to be used for imputation missing values while keeping randomness of the data.

* **Expectation Maximization Imputation: -** The EM or Expectation Maximization method of missing value imputation actually works on imputing dataset with expected values of missing numbers based on other observed values of the variables and then checks whether it was the most likely value to be imported or not. If not, then this iterative process continues until it reaches the most likely values. As it under estimate the standard error, so this imputation method is good if you are not bothered by high standard error and low p values of individual items.

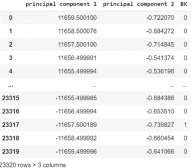
After comparing all the missing value imputation techniques, I have zeroed down to the imputed and standardized dataset using MICE as it keeps the statistical uncertainty without losing a single datapoint keeping the variance and bias of the dataset much in control and In good shape.KNN imputation follows MICE as the second best procedure of missing value imputation.

**4.3.3 Generate few new features using feature engineering: -**

Now after the data clearing, imputing and visualization, we introduced feature engineering and created additional columns/features from the existing one. It enhances the accuracy of machine learning model. After doing feature engineering we have a dataset with 23830 rows and 210 columns.

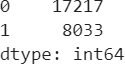
# 4.4 Machine Learning Modelling & Prediction: -

* **4.4.1 Test Train Split: -** After the data preprocessing and feature engineering, I have splitted my dataset into training data and testing data using 80:20 ratio.
* **4.4.2 Dimensionality Reduction Using PCA: -** Principal Component Analysis is an unsupervised machine learning technique that helps us to identify the pattern of the data on the basis of the correlation among the features. Due to having 210 features in the dataset post feature engineering, I used PCA-as it did find the directions of maximum variance in high dimensional data and projected it on a new orthogonal sub space with lesser or fewer dimensions/features than the original one. I did project all data in two principal components (or in 2D Dimension) successfully and thus it reduced my dimensionality effectively.



* **Dimensionality reduction using Linear Discriminant Analysis: -** Unlike PCA, Linear Discriminant Analysis uses class label to focus on finding a lower dimensional space for the data where there are class separabilities and training datas are mapped far apart. It is a supervised way of reducing dimension since it uses class label. LDA is well-suited for multi- class problems but should be used with care when the class distribution is imbalanced because the priors are estimated from the observed counts. Thus, observations will rarely be classified to infrequent classes.
* **Quadratic Discriminant Analysis: -** Quadratic Discriminant Analysis is a part of LDA where the covariance matrix for each each individual class in a dataset in computed. This method is useful if there is a particular knowledge of individual class covariance. Unfortunately, QDA cannot be used as a dimensionality reduction technique.
* **4.4.3 Oversampling of Imbalanced Dataset:-** Since my dataset is a highly imbalanced one with only 558 initial(136 after preprocessing) data point for “Yes” category and Rest of the whole dataset for “No” category ,naturally the whole dataset, even after data preprocessing and feature engineering, remains imbalanced and accuracy might be highly biased towards the “No” category since the “Yes” category data points are on lower side that this class will be projected as “No” class only. ML techniques such as Decision Tree and Logistic Regression are more bias towards the **majority** class and they tend to ignore the minority class. As a result, it creates major misclassification of minority target class compared to majority ones. In more technical words, due to imbalanced data distribution in target class of our dataset, it will have very lesser **recall**. To handle this problem, we use over sampling of the dataset using ADASYN Sampling, SMOTE Sampling procedure and check which one suits the best. I ignored under sampling as Near Miss Sampling as it reduces the number of datapoints in training dataset, resulting serious adverse effects on accuracy.
* **SMOTE (Synthetic Minority Over-sampling Technique)** is the one of the best over sampling techniques which creates synthetic yet not duplicate samples of minority class to make minority class equal to majority class. SMOTE does it by selecting similar records and altering it in one column at a time by random within the difference of nearest neighbors.
* **ADASYN (Adaptive Sampling Method for Imbalanced Data)** is another over sampling technique which works similar like SMOTE but it adds some random values at the end of the random lines in SMOTE which makes it more realistic and increases the variance of the model.
* On the other hand, the **Near Miss sampling** is another over sampling techniques where instead of resampling the minority classes, it resamples on majority classes and thus makes majority class equal to minority class using distance.

Though Near Miss sampling gave better accuracy than SMOTE, but the precision and recalls are much higher in SMOTE than in Near Miss as well as Near Miss is an under-sampling technique, I have gone ahead with SMOTE while doing the over sampling of imbalanced dataset. I could have proceeded with ADASYN, but according to our dataset, I proceeded with SMOTE.



# Balanced Target Class Algorithms after Oversampling

* **4.4.4 K Nearest Neighbors Classifier: -** KNN Algorithm is an unsupervised instance based lazy learning algorithm which can be used both for classification as well as for clustering. Here to classify the belongingness of a particular data point to a particular label, first it finds the K nearest neighbors of that target point and then with a majority voting scheme, the class which has higher representation in that nearest neighbor becomes the class where the target data point will belong too. Since KNN algorithm does not need to be learnt on training

data and spends most of the time on test data so computation of it becomes expensive when the dataset is really large as well as it produces low bias and high variance in test accuracy which is not acceptable.

* **4.4.5 Logistic Regression Classifier: -**Logistic Regression is a classification algorithm that is used to assign observation to a discrete set of classes based on their class belongingness probabilities using Logit function. Logistic Regression perform well when dataset is linearly separable but in high dimensional dataset it performs badly. Tough this algorithm is less prone to over fitting but it can overfit the data in high dimensional spaces. We can also see from our result without PCA dimension reduction the logistic regression accuracy was 65% but after doing the dimension reduction, the accuracy got increased to around 75%.
* **4.4.6 Decision Tree Classifier: -**A decision tree classifier is a simple classification algorithm where data continuously gets splatted according to a parameter. This algorithm consists of nodes, edges/branch and leaf nodes. In this classification, a new example is being classified by submitting it to a series of tests in order to determine the class label of the example. These tests are organized in a hierarchical structure known as decision tree. Decision Trees follow Divide- and-Conquer Algorithm. Although decision tree method is quick, very fast in classifying unseen data, excludes unimportant features, it is very easy for training data to get overfitted as well as small changes can result in massive changes in Decision Tree.
* **4.4.7 Random Forest Classifier: -** Random Forest Classifier is another supervised machine learning algorithm which creates decision tree on randomly selected data points. Then using the majority voting concept, it selects the best prediction among all the predictions that one has got from each of these decision trees. Random Forest is also very helpful in getting the feature importance, especially when dataset is really large. The best advantage of this classifier is that it is highly accurate and robust due to the numbers of trees that participate in the process. Additionally, it does not get affected by overfitting problem as it takes average of all prediction coming from all trees, which omits any chance of having bias. This is why it is one of the most popular ML algorithm in financial industry due to its high and balanced train as well as test accuracy.
* **4.4.8 Support Vector Machine Classifier: -**Support Vector Machine Classifier is a popular machine learning algorithm which finds optimal hyperplane to separate the data points having different class membership with highest margin which helps to classify new test data in the test dataset. SVM does not work well for large datasets because it takes high training time and it also takes more time in training compared to Naïve Bayes alongside working poorly with overlapping classes. Besides that, it is also sensitive to the type of kernel being used.
* **4.4.9 Ada Boost Classifier: -** Boosting algorithm is slowly becoming very famous in predicting bankruptcy or any other financial deciders due to its ability to combining multiple low accuracy models to create a really high accuracy for test data. Ada Boost is one of the interactive based ensembles based boosting classifier which combines multiple weak classifiers to get better and strong accuracy. This algorithm first selects a training subset randomly. Then in each iteration it trains the training dataset which is selected based on the accurate prediction of last iteration. Now in each step it adds weights to weak classifiers to make them strong as well as it adds weights to trained strong classifier in each next step.

. At last to classify, it uses ensemble-based voting method accorss all classifiers, which were built in each iterative phase, to select the best one among these. This algorithm is not prone to overfitting and fits the data accurately.

* **4.4.10 Xg Boost Classifier: -** Xg Boost or Xtreme Gradient Boosting is a decision tree based boosting algorithm which uses boosting method for machine learning modelling.This model takes feedback from previously run model by predicting the residuals of prior models and according to the feedback, it trains the dataset to minimize the error made earlier. This model already has built in regularization parameters which help it to be lesser prone to over fitting. Also, due to being run on parallel processing system, it works very fast even with large number of datasets.
* **4.4.11 Naïve Bayes Classifier: -** Naïve Bayes classifier is a probabilistic machine learning model that is used for classification task. The assumption is that the predictors/features are independent. In other words, presence of one particular feature does not affect the other predictor. Hence it is called naive. The classification is based on Bayes theorem where we find the conditional probabilities of the belongingness of a test data to a particular class based on the prior knowledge of some event. Another assumption of Naïve Bayes is that all predictor will have an equal effect on the outcome. The requirement of having the predictors as independent variables is one of the biggest drawbacks of this algorithm as this does not happen in real life. This hinders the performance of the naïve Bayes classifier algorithm.
* **4.4.12 Cat Boost Classifier: -** Cat Boost is a recently open-sourced machine learning algorithm from Yandex, Russian Internet Company. Cat Boost comes from Category Boosting. Unlike other traditional machine learning algorithms, it provides superb accuracy result without extensive data training. Apart from that, Cat Boost can handle the categorial data automatically without any explicit data preprocessing using a combination of statistical methods. Due to lesser dealing with hyper parameters, Cat Boost is lesser prone to over fitting.
* **4.4.13 Ensemble Method using Voting: -** A collection of several models working together on a single set in called Ensemble method. This method is lesser prone to over fitting, generates lowest errors, Voting is one of the easiest ways of combining multiple predictions from different machine learning models and by using soft or hard voting techniques it finds out which prediction from which algorithms gets the maximum votes and takes the one with highest votes as a final prediction class for the target values and corresponding algorithm as the final algorithm. Rather than being classified as a classifier this ensemble base devoting method is a wrapper which train the model as well as evaluate it parallelly to exploit different peculiarities of each algorithm. We can train data on different algorithm and predict the final output. In case of hard voting, the class and corresponding machine learning classifiers with highest number of votes will be chosen for the target value to be classified. Whereas, the soft voting, we use probability based weighted average value for each predicted class and each classifier and then we select the one, as winning class for the particular target value and corresponding classifier as winner, with highest weighted value of probability. A voting classifier is very useful to use when a there are multiple classifiers yielding high yet similar result and you want to ensemble all those classifiers to get better performance out of those classifiers in order to predict correct class belonging ness of a particular new unseen test data. Ensemble Method can be subdivided into four parts: -

Bagging, Boosting, Stacking, Cascading. Bagging tries to make predictions on small samples in our dataset and then it takes the mean of all prediction, which can reduce the variance error significantly, thus enhancing model performance smoothly. On the other hand, Boosting is an iterative procedure where weight on the observations gets adjusted based on the last classification to reduce the error. If any observation is classified incorrectly, then boosting can help it by assigning higher weight to it and at the same time by lowering the wright of the training data. This is why it is least prone to over fitting and gives accurate results.

# 4.5 Evaluation of Model and Validation: -

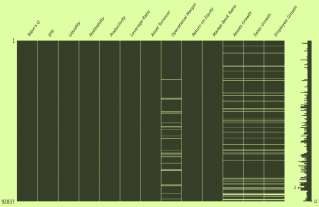
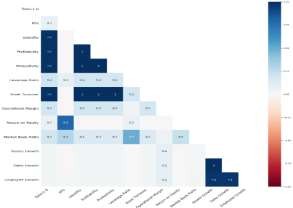
* **4.5.1 Cross-Validation: -** Cross Validation is the one of the procedures which prevents the model to be over fitted resulting into high accuracy on training dataset and low on testing dataset. In this method we divide training dataset into a particular number of validation subsets only. This number is a hyper parameter K which is to be decided according to the specific problem but in general we take it as 10.then we keep a hold on one of these subsets and train our model on the other remaining subsets and then subsequently check our test prediction mean scores on the hold on dataset. This process continues for K number of subsets until we optimize the over fitting tendency of the training dataset prior to testing it on new unseen dataset.
* **4.5.2 Precision, Recall, F Score: -** Precision refers to the ratio percentage of actual true positive prediction of the correct class belongingness of a particular new test dataset relative to the total number of positive correct prediction being made. It can be represented as **TP/(TP+FP).** Precision decides the total number of cases our model said was relevant were actually relevant. Whereas the recall is the ratio percentage of total number correctly predicted classes relative to all positive cases. It can be represented as **TP/(TP+FN).** With the metric, our intuition tells us that we should maximize the positive classes, known in statistics as recall, or the ability of a model to find all the relevant cases within a dataset. For a model to be highly accurate on test dataset, the precision as well as recall have to be as high as possible, close the values are to 1, better the overall accuracy is. F score comes into picture to deal with different machine learning models with low bias and high variance or vice versa. F score is the (1+ beta) ^2 times the harmonic mean of precision and recall. In other words, F score is **(1+beta) ^2\*((2\*recall\*precision)/ (precision+ recall))**. The closer the F score to 1, the better the model performance on new unseen test data set will be. In order to have a classification model with the desired balance of recall and precision, we maximize the F1 score as much as possible.
* **4.5.3 ROC AUC Curve: -** The ROC OR Receiver Operating Characteristic curve plots true positive rate against false positive rate as a function of the model’s threshold to classify correctly a positive case. the ROC curve shows the way relationship between recall and precision changes due to varying threshold for identifying a positive in our model. ­­­

The threshold represents a particular value above of which a data point is considered to be in the positive class. In Whereas the AUC is the area under the ROC curve which helps us to calculate the overall measure of a classification problem based on the area under ROC corbeau falls under 0 to 1 and higher the AUC value, better the classification accuracy is.

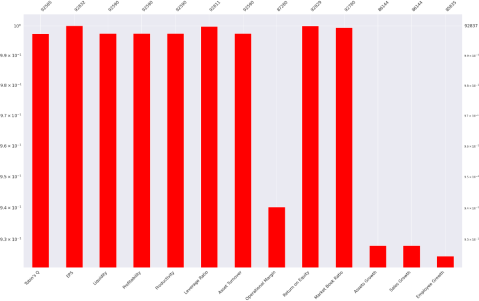
* **4.5.4 Confusion Matrix: -** Confusion matrix is a performance management matrix for machine learning two or more classes’ classification. It is the four different combinations of predicted and actual values. The rows represent predicted values and the columns represent actual values. True Positive refers to the value that is predicted as true and actually those are true. False Positives (Type I Error) are those cases which have been predicted as yes but actually no. True Negative refers to those cases where the value is predicted as no and actually those are no. False Negatives (Type II Error) are those cases where the value is predicted as no, but those are actually yes. Accuracy of a model can be calculated from confusion matrix and that is (Total Positive+ Total Negative)/Total.
* **4.5.5 Cumulative Gain Chart: -** The cumulative gain chart is a classification model performance evaluation chart where beyond ROC-AUC, we check the model performance as this chart is very much easier to visualize for a no technical person than ROC-AUC. It compares the result to the with any random pick. It shows percentage of positive results achieved when we are considering a certain percentage of population with high probabilities to target according to the model**.**

# Results

# 5.1.1 Missing Values and Dataset Visualization: -



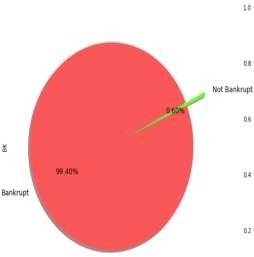
**Correlation Among Missing Values Sparsity Matrix of Missing Values**



**Missing Value Representation Bar Plot**



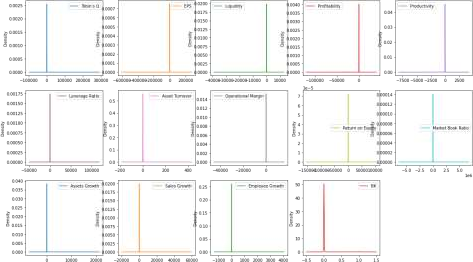
# Year Wise Total Number of Bankrupt Companies Plot (We can see there were a lot of bankrupt companies in the dataset between 1985-1995 whereas around 2008- 2010 also has seen bankruptcy cases due to 2008 recession)



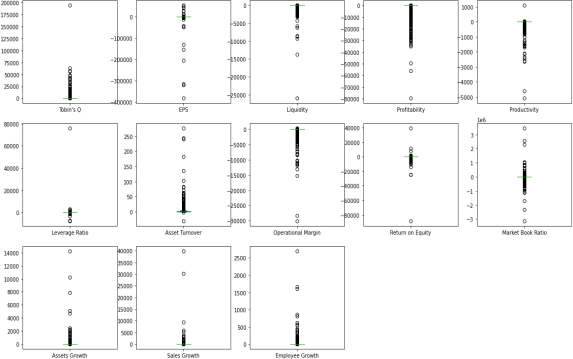
**Pie Chart to show the initial imbalance state of target class (Bankrupt or Not).**



**Initial Numbers of Missing Values Count of Missing Values after MICE Imputation**

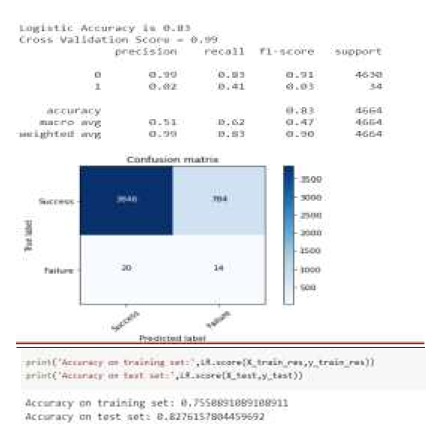
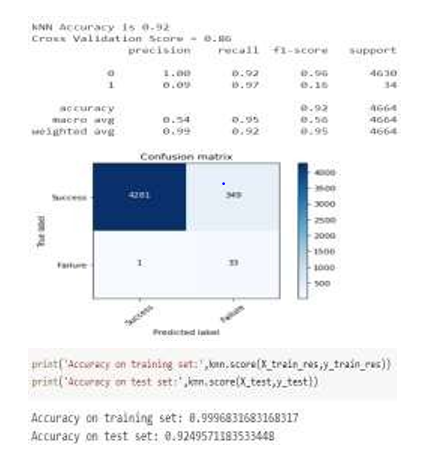


**Density Plot of Different Features**

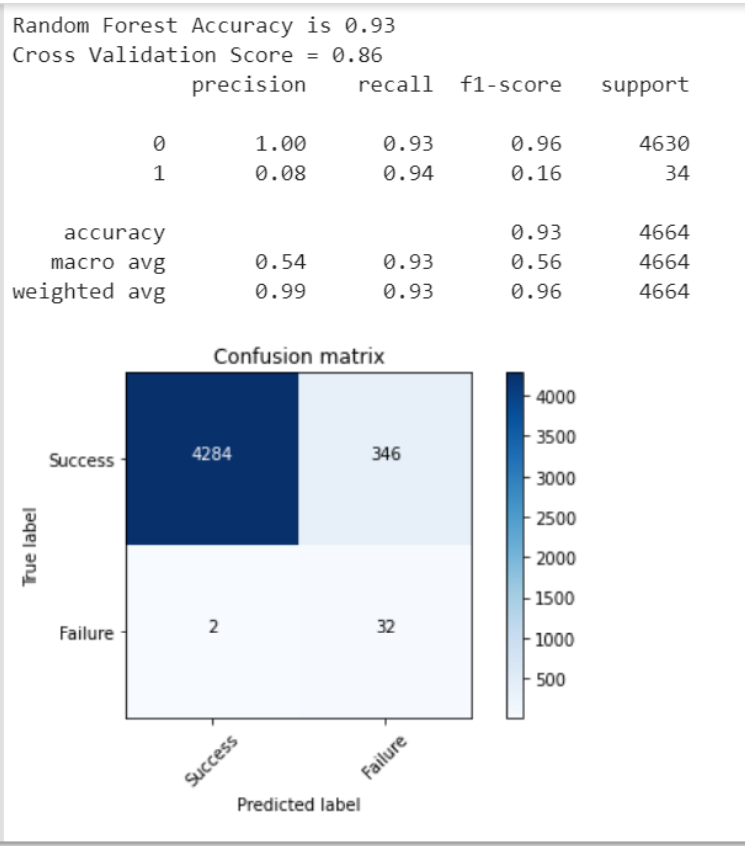
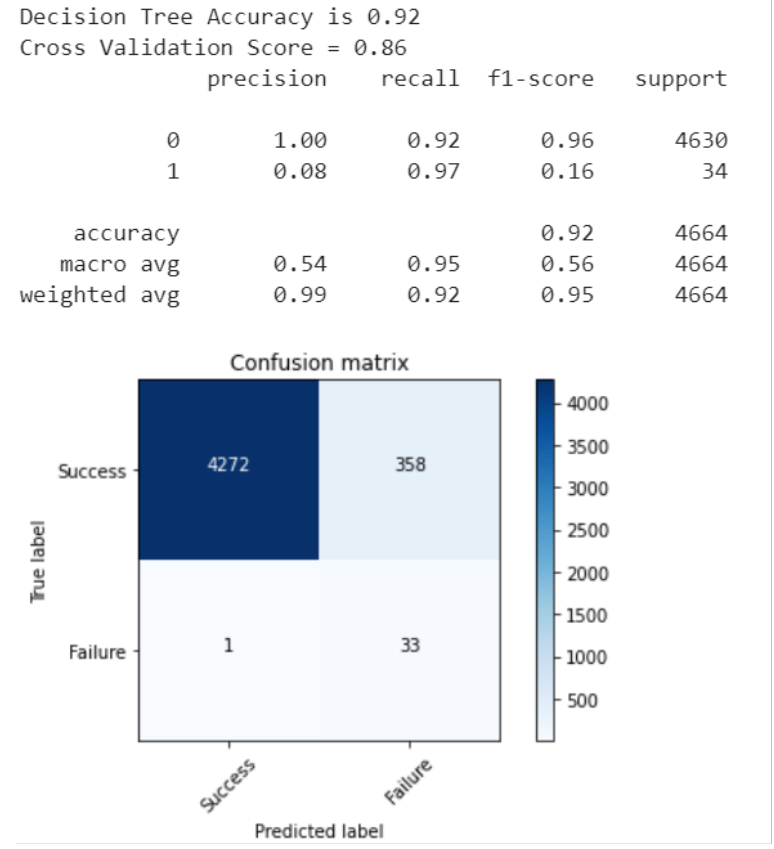


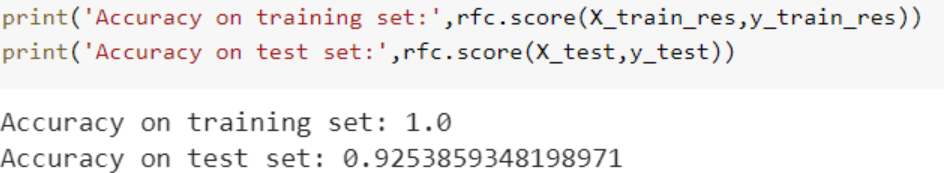
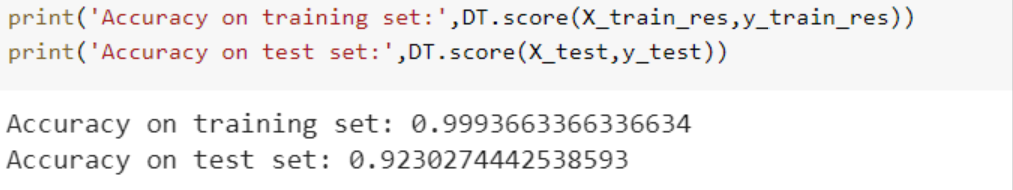
**Box Plot of Various Features in the Dataset**

**5.1.2 K Nearest Neighbors Results 5.1.3 Logistic Regression Results**

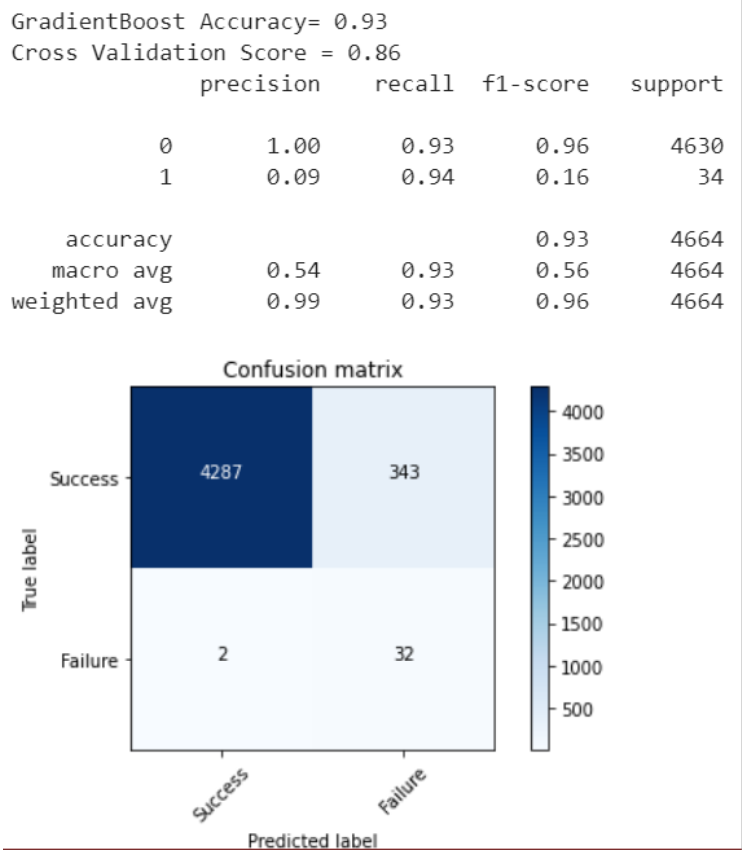
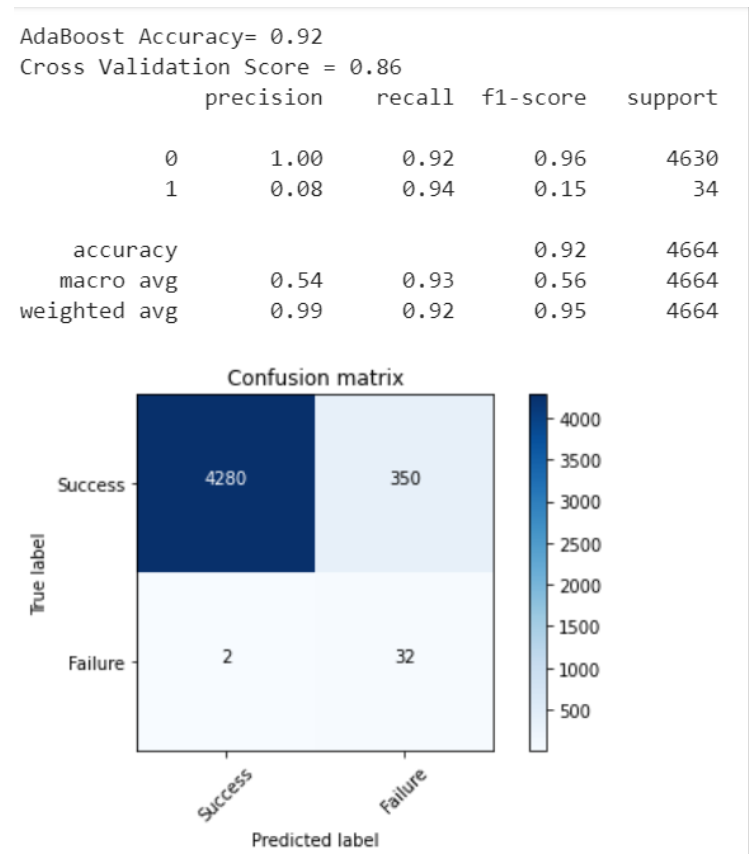


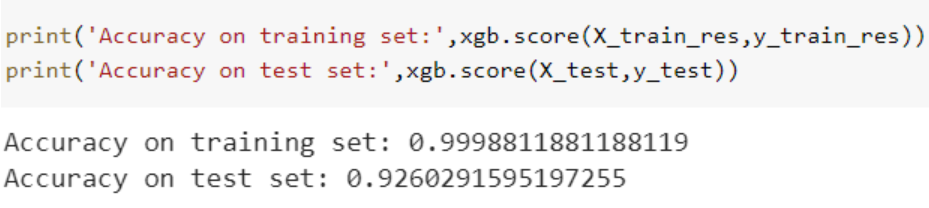
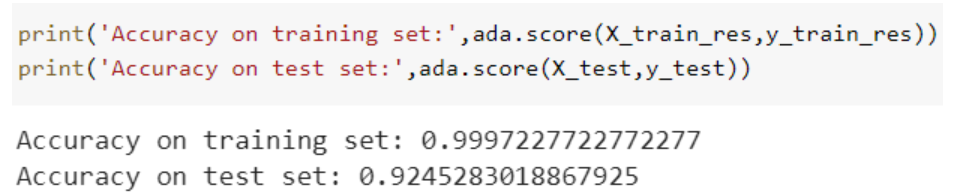
**5.1.4 Decision Tree Results 5.1.5 Random Forest Results**



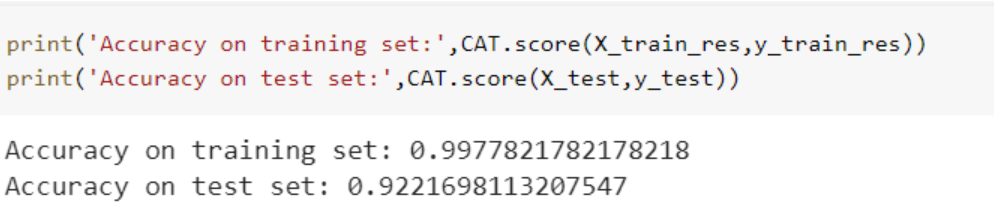
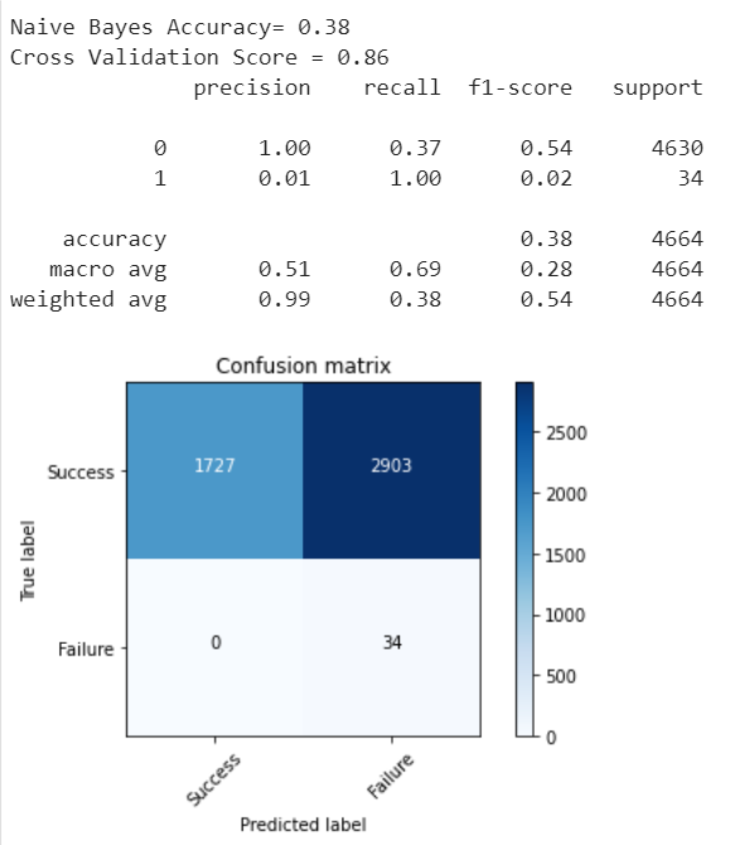


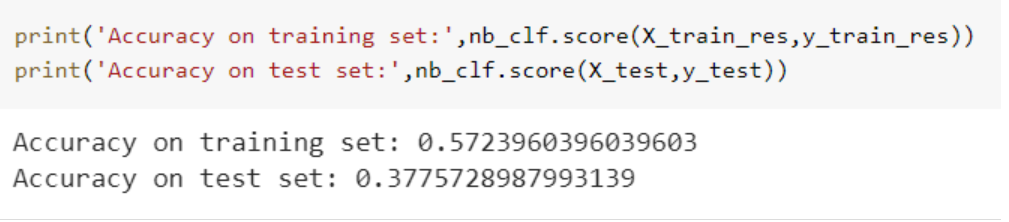
**5.1.6 Xgboost Result 5.1.7 Adaboost Results**

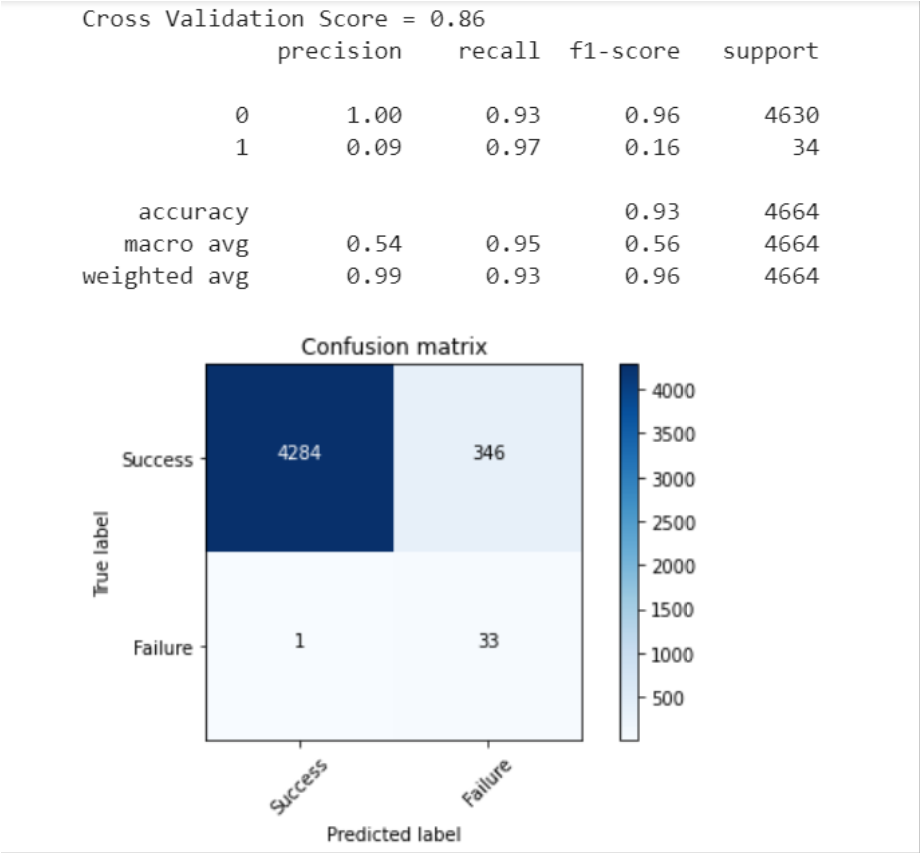
 

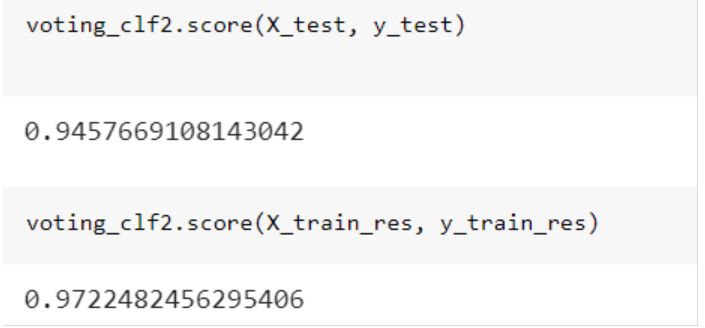
**5.1.8 Naïve Bayes Results 5.1.9 Cat Boost Results**





**5.1.10 Ensemble Based Voting Method Results 5.1.11 SVM Result**

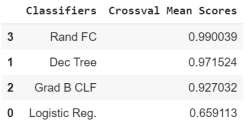
 



**5.1.12 Comparison Result**

**5.1.13 Cross Validation Mean Score Comparison**

**Pic-1 Pic-2**

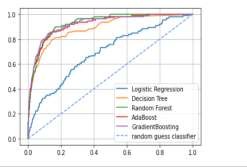
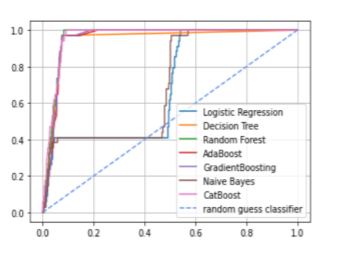


**Pic 1: -Cross-Validation Mean Score for all Classifier before Feature Engineering and PCA of the Over Sampled Balanced Dataset**

**Pic 2: -Cross-Validation Mean Score for all Classifier after Feature Engineering and PCA of the Over Sampled Balanced Dataset**

**5.1.14 ROC AUC Curve Result**

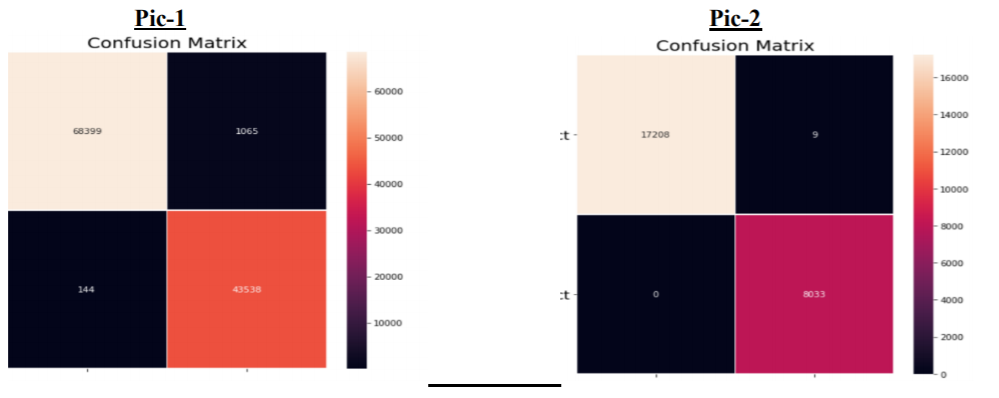
**Pic-1 Pic-2**

**Pic 1: -ROC-AUC Curve for all Classifier before Feature Engineering and PCA of the Over Sampled Balanced Dataset with with Highest AUC value is 0.82 for Random Forest &Boosting Algorithms**

**Pic-2: -ROC AUC Curve (After Feature Engineering and PCA along with Over Sampling of the Dataset) For All Classifier with Highest AUC value is around close to 1 for Random Forest and Boosting Algorithms.**

**5.1.15 Confusion Matrix Result Comparison**



# Pic 1: -Overall Confusion Matrix of the Over Sampled Balanced Dataset before doing Feature Engineering and PCA Dimension Reduction

**Pic-2: -Overall Confusion Matrix of the Over Sampled Balanced Dataset after Doing Feature Engineering and PCA Dimension Reduction**

Each **row** of the 2\*2 **confusion matrix** represents the instances of an actual class (Actual No for first row and Actual Yes for second row as per our case) and each **column** represents the instances of a **predicted** class. (Predicted No for first column and Predicted Yes for second column as per our case).

# 5.1.16 Overall Precision, Recall and F Score Values Comparison



**Pic-1: -Overall Precision, Recall and F Score of the Over Sampled Balanced Dataset before doing Feature Engineering and PCA Dimension Reduction**

**Pic-2: -Overall Precision, Recall and F Score of the Over Sampled Balanced Dataset after doing Feature Engineering and PCA Dimension Reduction**

3

**6. Discussion**

**6.1.1 Comparison to the Literature: -** After seeing the above results, if we compare it with my literature review then we can easily see due to advancement of modern machine learning and artificial intelligence-based technologies, you can achieve much better results with boosting algorithms from what it used to be back in 1970s. Most of the algorithms mentioned in my literature review have been around Z score, time series modelling, Random Forest, Logistic Regression only. Earlier the dimension reduction techniques PCA to achieve more accurate results were not common, but with my project I have introduced the application of boosting machine learning algorithms, as well as modern day techniques like SMOTE for over sampling of imbalanced target class, Principal Component Analysis for dimensionality of features’ reduction and all these have given me an amazing accuracy in predicting whether a new company will be bankrupt or not in advance. As you can see from the above results, Random Forest has been consistent top classifier throughout the training set, cross validation set and testing set. Modern day Boosting Algorithms like AdaBoost, XgBoost and CAT Boost have also achieved as amazing accuracy as Random Forest. From our confusion matrix we can see, only 9 corporations out of all, have been misclassified as bankrupt when they are actually not. With false positive rate (FP/(FP+TN) as low as 0.05 and false negative rate (FN/(FN+.TP) of zero, it shows how strong the boosting algorithms, along with Random Forest, can be in predicting financial instances from large scale datasets compared to other traditional methods. So, I believe there is probability of having huge success rate in accurately predicting bankruptcy if we frequently keep applying these modern-day Boosting Machine Learning technologies to our model.

**6.1.2 Discussion of Dimensionality: -**The featured engineered dataset suffers from the curse of dimensionality as if we use too many features compared to the number of observations, the risk of the dataset to get overfitted massively resulting a very poor result on test set is obvious. The method of PCA imposes unsupervised learning to reduce the dimensions through eigen values and eigen vectors and by projecting high dimensional spaces orthogonally to low dimension spaces. The dimensionality reduction techniques from PCA performed with our expectations, followed by LDA and finally QDA. One of the possible explanations of inferiority of LDA and QDA when compared to PCA is that these two models find it difficult to select reliable decision boundaries seem to perform poorly, which indicates that QDA is not optimal for predicting bankruptcies. For the PCA, our ensemble weighted probability voting results show that Random Forest, along with gradient boosting algorithms like Adaboost, XgBoost, CatBoost is better for processing a lot of features, due to the de- correlation procedure that the model applies on each subtree. This is indicated in the RF model and boosting models yielding the highest TPs (Around 4285 for each algorithm) while keeping FPs as low as around 350 in each algorithm. Decision Tree also perform excellent on the balanced dataset with

high dimension but since Random Forest is a collection of many decision tree, it produces a robust

accuracy result on test dataset which is very less prone to overfitting resulting in low bias and higher variance of the model.

**6.1.3 Comparison of Variable Selection: -** We noticed that the Tobin’s Profitability, Productivity, Leverage ratio, Return on Equity and Liquidity measures are recurring with an around 21% total explanation of the problem (See Random Forest feature importance above). Hence, we assume these apparently are good predictors and thus higher the chances that they would improve bankruptcy predictions. Other types of financial variables, deemed important in our models are Operational Margin, Market Book Ratio. This is interesting, since ratios for liquidity, like CATA (Current assets/Total assets), like NITI (Net income/Total assets), like TDTA (Total debt/Total assets), like TLTA (Total liabilities/Total assets) and like EBIT/TA (Earnings before interest and taxes/Total assets) are highly used in the past. Hence, our results show that these measures are valid to some measures. This indicates that the previous intuition about financial ratios seemed to be somewhat correct, in terms of my selected ratios. The previously used ratios are primarily dependent on assets, whereas our results also show that assets are much more important. Liabilities comes next. More than 60% of the features are related to assets. This indicates that assets have a higher predicting power of bankruptcies than liabilities keeping in tune with the previous finding.

**6.1.4 Discussion on Applications:-** If application of our Bankruptcy prediction model based on machine learning should have any real world practical value, then it can be said with confidence that these models can be very advance to predict the bankruptcy ahead of time which actually may help the firms as well as the investors and stake holders to be prepared for such kind of mishaps well in advance. Since the dataset was a yearly dataset, so our model as of now can predict the bankruptcy status one year in advance. From a Bank managers’ perspective while giving bank loan, that would be sufficient to know one year in advance but from the stake holders’ and investors’ perspective a one financial year prior or current financial year prediction is not sufficient as they deal with huge amount of investment they already made. So, in that area, I need to work now more to make it a multi- year’s advance bankruptcy prediction model.

**6.1.5 Further Research:-** Few of the further research in this area that I want to continue and to implement in future are creating a multi-year bankruptcy prediction model which, instead of predicting bankruptcy only one financial year prior or based on current whole financial year performance , can predict potential bankruptcies even 3 year prior in advance. Another area, I want to work on is that, if there is any underlined relation between what the firm posts on social media and bankruptcy chances based on Natural Language Processing, Text Mining and Sentiment Analysis. Last but not least, I want to work on also to research about the effect of credit rating and sudden massive layoffs on a particular firm, specially a well-funded early stage starts up, in going to potential bankruptcy phases. Additionally, I want to check whether the recent deep learning frame works and algorithms can predict bankruptcy in a better way than machine learning or not. I am also eager to apply grid search for regularization and tuning of different hyper parameters in different ML models to get more better accuracy even with an improved and better version of dataset as a part of next steps.

# 7. Conclusion

The field of bankruptcy prediction has been evolving since the earliest adoptions at the beginning of the twentieth century. The field has gone through a reformation as computational power has excelled. The most renowned and applied models today are simple ratio models which were developed in the last part of the twentieth century. I have created nine different machine learning models, one for each of the techniques, with different statistical properties. The optimization of the models is done through maximizing of correct predictions of true positives, while consequently minimizing false negative predictions. My research indicates that Random Forest, XgBoost, AdaBoost & CatBoost are the four best performing models, obtaining a test accuracy for bankrupt companies at 92.5%, 92.6%, 92.45% and 92.21% respectively while achieving a ~ 99% training accuracy across dataset. Whereas, Random Forest and Gradient Boosting algorithms top the cross validation mean scores while validating the overfitting of dataset, which means these boosting machine learning models along with Random Forest are least prone to overfitting among all 9 algorithms and can predict any new test dataset quite accurately with low bias and low variance. Overall, these models obtain an AUC value of more than

* 1. with Random Forest AUC being the highest among all at 0.82(The AUC is a measure of degree of separability between target classes which in technical word is how good the model is to separate the target classes to their proper class categories). This notifies that we can correctly predict more than 4/5 of both bankrupt and non-bankrupt companies, or in other words that more than 80% of the variation in bankruptcies can be explained by our data at this threshold. Since, this was initially a problem of assymtratic target class distribution problem , so F score will be a great measure of accuracy and here F score is 0.9994 for the lesser dimension dataset and even ~0.98 with the high dimension dataset without PCA.This shows that our model has fit perfectly without over fitting or under fitting. Lastly, I did perform an ensemble-based majority voting classifier and found that maximum votes have gone for Random Forest and Gradient Boosting algorithms. These four best models are superiors compared to comparable models built on balanced class data. Logistic Regression accuracy is 75% and thus unacceptable. Naïve Bayes gives very less training and testing accuracies with ~57% and ~37% respectively. So, we can safely assume that this algorithm is the worst in predicting bankruptcy or any financial events. KNN and SVM are memory intensive, trickier to tune due to being instance-based by not being trained during training phase and the importance of picking the right kernel respectively, and don't scale well to larger datasets despite having some good accuracy scores. Currently in the finance industry, random forests are usually preferred over KNN and SVM's.Boosting Algorithms, specially Xg Boost and CatBoost, should be regularly used in financial industry in coming future along with traditional favorite Random Forest to predict bankruptcy even in a better manner Our results notifies that even with the cross validation and PCA the performance decay is too large for KNN and SVM to compete with Random Forest and modern Boosting Algorithms.

# 8. Reference

**Tomasz Koro**, EVALUATION OF THE FACTORS INFLUENCING BUSINESS

BANKRUPTCY RISK IN POLAND. Financial Internet Quarterly „e-Finanse” 2017, vol.13/ nr 2, s. 22-3.

**Frank Wagenmans and Dr. A. J. Feelders**, Machine Learning in Bankruptcy Prediction. ICA- 3870154. July 3, 2017.

**Håkon Kambestad Sundal & Karoline Hatlestad , Supervisor: Assistant Professor Kai Liu** , Factors Affecting the Probability of Bankruptcy, NORWEGIAN SCHOOL OF ECONOMICS.

**James Ming Chen**, Models for Predicting Business Bankruptcies and Their Application to Banking and to Financial Regulation.

**Li Yuan** ,Implementation of Self-Organizing Maps with Python Implementation of Self- Organizing Maps with Python, University of Rhode Island.

**K Venkatanarayana#1, Dr B.Kondaiah\*2, VVR Manoj@3** ,A Survey on Machine learning techniques for Bankruptcy prediction, #1 Computer Science and Technology, Sri Krishnadevaraya University, Ananthapuramu,India \*2 Computer science and Technology, Sri krishnadevaraya University, Ananthapuramu,India @3Dhanekula Institute of Engineering and Technology, Vijayawada, India.

**S. Kotsiantis1, D. Tzelepis1, E. Koumanakos2 and V. Tampakas1** ,Efficiency of Machine Learning Techniques in Bankruptcy Prediction, 1Department of Accounting, Technological Educational Institute of Patras, Greece 2National Bank of Greece, Credit Division, Greece

**BJÖRN MATTSSON & OLOF STEINERT**, supervised by ANDREAS DZEMSKI, CORPORATEBANKRUPTCYPREDICTIONUSING MACHINELEARNINGTECHNIQUES,

Department of Economics UNIVERSITY OF GOTHENBURG SCHOOL OF BUSINESS ECONOMICS AND LAW Gothenburg, Sweden 2017.

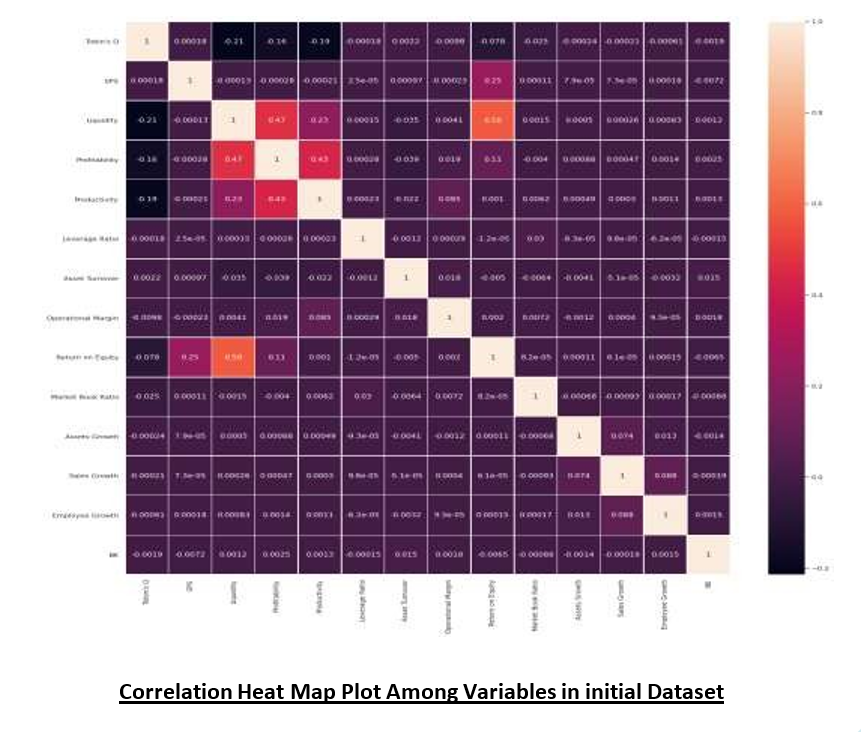
**Xavier Brédart1** ,Bankruptcy Prediction Model Using Neural Networks, 1 Warocqué School of Business and Economics, University of Mons, Mons, Belgium Correspondence: Xavier Brédart, Warocqué School of Business and Economics, University of Mons, Place Warocqué, 17, 7000, Mons, Belgique.

**Agostino Di Ciaccio1 and Giovanni Cialone2** ,INSOLVENCY PREDICTION ANALYSIS OF ITALIAN SMALL FIRMS BY DEEP LEARNING, 1Dept. of Statistics, university of Rome, La Sapienza, Italy 2Senior partner of Kairos Advisory srl., Italy.

# 9. Appendix and Annexure

* + - **A1) Technical Notes on Bankruptcy:-** When a company is about to go bankrupt , three important things might happen. Incase of inactive companies, it can disband without legal repercussions. In the second option, the court has the power to diaband any company due to neglect. This can place itself through missing information about the company or inadequate company structures. Lastly, the company can be vanished due to bankruptcy. If we want to describe bankruptcy is technical terms, that would be the scenario when overall liabilities have surpassed the overall assets and a firm is unable to cover the differences from operational income. In other way, the firm does not have liquidity to cover up any kind of short-term debt and due to that becomes insolvent.

# A2) Correlation Matrix: -



* **A3) Automated Feature Engineering: -** Feature engineering is the process of creating new features from the existing features in a dataset in order to increase the diversity and random ness in the dataset as well as to reduce bias towards some particular features. So here we have 14 financial variables and using feature tools in python I have created different ratios and other variables that might be of importance from financial perspective. The performance of the predictive model is mostly dependent on the quality of the features in the dataset being used to train that model. If you are able to create new features helping in providing more information to the model about the target variable, performance will keep getting better. Hence, when we don’t have enough quality features in our dataset, we have to lean on feature engineering, which was the case in my dataset. I automated the feature creation using python’s feature tools library. It has three major components, - Entries, which can be considered as a panda data frame. Deep Feature Synthesis, which created new features from existing one using feature primitives like addicting, subtraction, multiplication, division etc. Also, another advantage of feature tool is since it is based on primitives, it can be explained even to non-technical person. So, using this tool, I have created 219 features in the dataset from the existing 15 features.

# A4) Mathematics behind Top Performing Machine Learning Algorithms: -

* **Mathematical Steps for PCA: -**

The **PCA algorithm** works in 4 simple steps: -

* Subtract each variable of the data with its mean and then store that result in a Row Adjust Data variable. (Our assumption is that the data is of the shape (m, n), where m being the number of instances and n being the number of variables.
* Then we will calculate the co variance matrix of the dataset.
* After that we will calculate the eigen values and eigen vector and then we will select the eigen vectors with highest eigen values as the principal components of our dataset.
* Then
* we will create a new feature vector variable to transform our dataset and store it in this vector. Data transformation to lower dimension is done by multiplying this feature vector by the initial Row Adjust Data Variable. Resultant vector of transformed dataset would be of (n, k) shaped matrix where k is the number of principal components our dataset got reduced to and it is less than n.

# Mathematical Steps for SMOTE: -

The **SMOTE algorithm** works in 4 simple steps: -

* First, we select the minority class and then input vector associated with it.8
* Then we find the K nearest neighbors of each data points in imbalanced class in dataset.
* After that, we select choose any one of these neighbors randomly and place a synthetic point anywhere on the line that is joining the synthetic point and the point we selected in one of the nearest neighbors.
* We keep repeating these steps until the dataset or particular class of the dataset in balanced.

# Mathematical Steps for Random Forest Classifier: -

The **Random Forest algorithm** works in 5 simple steps: -

* First select 'k' features randomly from a total of 'm' features where k << m.
* Now out the 'k' features, calculate the node D using the best split point.
* Split the node into daughter nodes using the best split.
* Keep repeating steps two and three till when leaf nodes are finalized.
* Build forest by repeating above steps for 'n' times to create 'n' number of trees.

# Mathematical Steps for AdaBoost Classifier: -

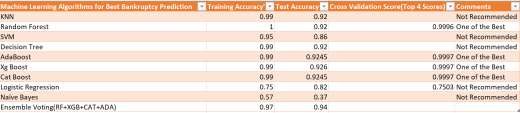
* The **Ada Boost algorithm** works in 4 simple steps: -
* When particular dataset with two class classification problem is given with -1 assigned to the negative class and +1 assigned to the positive class, this algorithm initiates weights at each point in the dataset where weight as **w=1/n**, where n is the number of datapoints.
* Then it runs an iteration algorithm equal to the number of wear classifiers a particular problem set is having (denoted by M) and initially fits weak classifiers to the dataset and then selects the one with lowest classification error. Then it calculates weight for the Mth weak classifier as theta=1/2\*log (1-weighted classification error)/ weighted classification error. For any classifier with accuracy higher than 50%, the weight is positive. The more accurate the classifier, the larger the weight. While for the classifier with less than 50% accuracy, the weight is negative. It additionally also keeps those classifiers making accuracy less than the random guess to take their contribution into consideration while doing the final prediction.
* Then it keeps updating the weight for each data point as **weight at (m+1)th iteration= (weight at mth interation \* exp(-theta\_m\*y\_i\*f\_m(x\_i)))/Z\_m** where Z\_m is a normalization factor that ensures the sum of all instances of weight will be equal to one,f\_m(x\_i) is the mth weak classifier and theta\_m is the corresponding weights assigned to mth weak classifier a misclassified case is from a positive classifier then the value of the exponential part would be always greater than one. This class would be updated with a larger weight after each iteration. Same happens with if misclassified class is from negative classifier.
* After m th iteration the final prediction is obtained as the weighted sum of all predictions from each weak classifier and it can be represented as F**(x) =sign (sum over (m=1 to M(theta\_m\*f\_m(x)).**

# Mathematical Steps for Cat Boost Classifier: -

The **CAT Boost algorithm** works in 4 simple steps: -

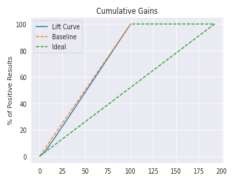
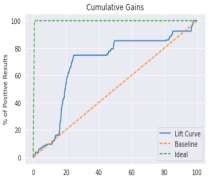
* Its firsts initialize the model with a constant value.
* Then it runs m iterations based on the number of weak classifiers to calculate the pseudo residuals denoted by an augmented minimum matrix of a differentiable log loss function.
* Then it trains the training dataset using this pseudo residual values.
* After that it compares multiple such initialized models with constant values from each integration and update the model accordingly to get the final prediction.

# A5) Accuracy Table for all classifiers: -



**A6) Cumulative Gain Chart: -**

**Pic-1 Pic:2**

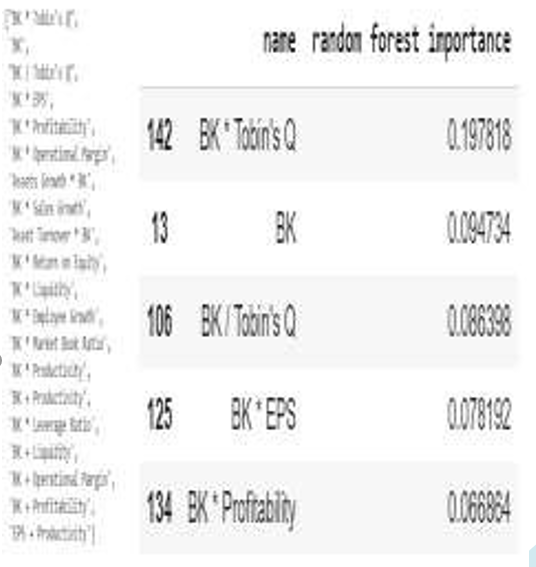
# Pic-1: - Cumulative Gain and Lyft Chart before doing Feature Engineering and PCA

**Pic-2: - Cumulative Gain and Lyft Chart after doing Feature Engineering and PCA**

So, from the first (Pic-1) cumulative gain chart, we can say only 20% positive results of accurately bankruptcy prediction was achieved when we were considering 25% percentage of population with high probabilities to target according to the model**.** In contrast to that**,** when we do feature engineering to our dataset as well as convert imbalanced class data to balanced one,60% positive results of accurately bankruptcy prediction was achieved when we considered only 20% percentage of population with high probabilities to target according to the model. That is huge improvement of the predictive model.

# A7) Variable Importance: -

**Pic-1 Pic:2**

**Pic-1: -Random Forest Feature Importance before doing Feature Engineering Pic-2: - Random Forest Feature Importance after doing Feature Engineering**

We noticed that the Tobin’s Profitability, Productivity, Leverage ratio, Return on Equity and Liquidity measures are recurring with an around 21% total explanation of the problem (See Random Forest feature importance above) in the feature engineered dataset. Hence, we argue these apparently are good predictors and thus most likely would improve bankruptcy predictions.

# A8) Computer Speciﬁcations: -

My analysis is completed using google colab python notebook with the version 3.7 of Python.

Running under: Windows 10 x64 and Google Drive Cloud. Intel(R) Windows 10 CPU, RAM: 4 GB

# Thank You