Project Report: Bankruptcy Prediction

Presented by:

- ***Yeshwanth Vemula -*** yvemula1@hawk.iit.edu
- ***Sundar Machani – smachani@hawk.iit.edu***

Table of Contents

1. Introduction
    1.1 Background
    1.2 Objective

2. Data
    2.1 Dataset Description
    2.2 Data Preprocessing
    2.3 Data Stability and Exploration

3. Model Training
    3.1 Logistic Regression (with Feature Selection)
    3.2 Random Forest
    3.3 XGBoost
    3.4 KNeighbourClassifier
    3.5 Deep Learning (MLPClassifier)

4. Model Evaluation
    4.1 Cross-Validation
    4.2 Performance Comparison

5. Potential Improvements
    5.1 Identified Pitfalls
    5.2 Proposed Improvements
    5.3 Additional Experiments

6. Code and Dataset

7. Conclusion

1. Introduction

1.1 Background

Financial stability is a cornerstone for any business's success, and the ability to predict potential bankruptcy is crucial for investors, creditors, and the company itself. Bankruptcy prediction involves assessing a company's financial health, identifying early signs of distress, and taking preventive measures. Traditional financial analysis methods may not always capture subtle patterns and interactions within the data, making machine learning an ideal tool for such predictions.

In this context, the project delves into the application of advanced machine learning techniques to enhance the accuracy and efficiency of bankruptcy prediction. By leveraging the power of algorithms, we aim to unearth intricate relationships within financial datasets that might elude conventional analysis. This project addresses the growing need for sophisticated tools in financial analytics to ensure timely and informed decision-making.

1.2 Objective

The primary objective of this project is to harness the capabilities of machine learning algorithms for the accurate prediction of bankruptcy based on financial data. The goals encompass:

1. Exploration of Diverse Models: Investigate the performance of various machine learning models, each with its strengths and weaknesses, to gain a comprehensive understanding of their suitability for bankruptcy prediction.

2. Development of Accurate Models: Strive to build predictive models that are not only accurate but also reliable. The focus is on achieving a balance between precision and recall, crucial metrics in the context of bankruptcy prediction.

3. Algorithmic Diversity: Evaluate a spectrum of machine learning algorithms, including Logistic Regression, Random Forest, XGBoost, KNeighbourClassifier, and Deep Learning (MLPClassifier). The exploration of diverse algorithms aims to identify the most effective approach for the specific challenges posed by bankruptcy prediction.

4. Robustness and Generalization: Ensure that the developed models exhibit robust performance on both training and unseen test data. The emphasis is on creating models that generalize well to real-world scenarios, enabling their practical applicability.

5. Model Interpretability: Strive to interpret the decisions made by the models to enhance transparency. Understanding the features contributing to predictions can provide valuable insights into the financial indicators influencing bankruptcy predictions.

In summary, this project is a comprehensive exploration of machine learning techniques for bankruptcy prediction, with the goal of delivering models that contribute to informed decision-making in the realm of financial analysis.

2. Data

2.1 Dataset Description

The dataset utilized in this project, named 'data.csv,' serves as the foundation for bankruptcy prediction. It encapsulates financial attributes, with a binary target variable 'Bankrupt?' denoting the bankruptcy status of companies. The dataset undergoes preprocessing to address categorical variables, standardize numerical features, and tackle class imbalance through Synthetic Minority Over-sampling Technique (SMOTE).

2.2 Data Preprocessing

Label Encoding: Categorical variables are encoded using label encoding to facilitate their incorporation into machine learning models.

Feature Scaling: Numerical features are standardized using StandardScaler, ensuring that all variables contribute uniformly to the model.

Handling Class Imbalance with SMOTE: The imbalanced distribution of the 'Bankrupt?' classes is mitigated using SMOTE, a technique that generates synthetic samples of the minority class to achieve a more balanced representation.

Attribute Information

The dataset comprises a comprehensive set of financial features (X1 to X95) that encapsulate various aspects of a company's performance. These attributes include return ratios, profit margins, growth rates, liquidity ratios, leverage indicators, and more. Each feature provides a nuanced perspective on the financial health and operational efficiency of the companies under consideration.

Understanding the specifics of each feature is crucial for model interpretation and informed decision-making. For instance, features like 'Net Income to Total Assets' (X86) and 'Current Asset Turnover Rate' (X71) offer insights into profitability and operational efficiency, respectively.

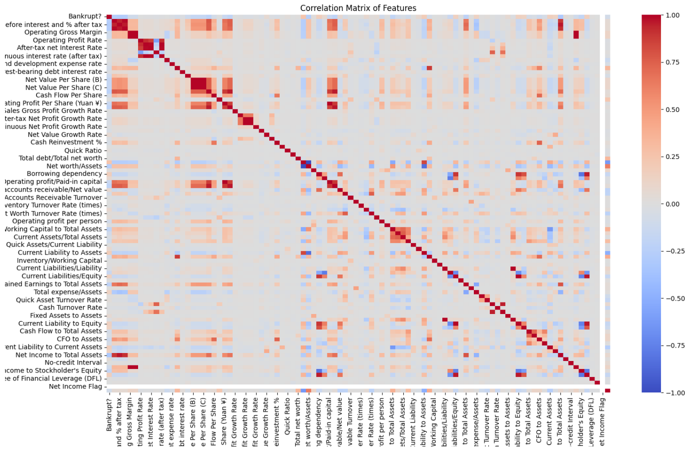
The target variable 'Bankrupt?' (Y) is binary, indicating whether a company is prone to bankruptcy or not. This binary classification problem forms the crux of the predictive modeling task.

In summary, the dataset's richness lies in its diverse financial attributes, providing a holistic view of companies' financial landscapes. The preprocessing steps ensure that the data is ready for the application of machine learning algorithms, fostering the development of robust bankruptcy prediction models.

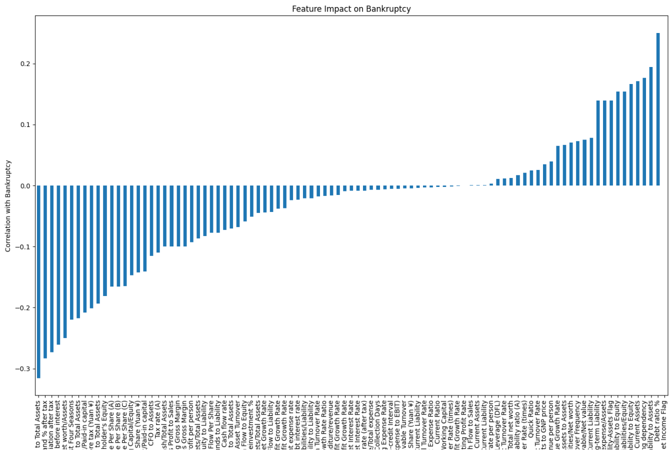
2.3 Data Stability and Exploration

The dataset is examined to understand the distribution of stable and unstable (Bankrupt) instances. As of the current state:
Stable Instances: 96.77372048687491 % of the dataset is stable.
Unstable Instances: 3.2262795131250916 % of the dataset is unstable/Bankrupt.

Correlation Analysis
A correlation matrix is computed to investigate the relationships between different features. The matrix is visualized using a heatmap:



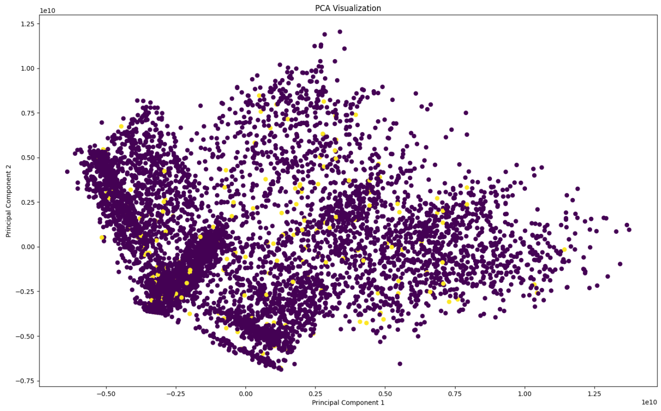
Feature Impact on Bankruptcy
The correlation between individual features and the target variable 'Bankrupt?' is analyzed. The impact of features on bankruptcy is visualized through a bar chart:



Dimensionality Reduction Techniques

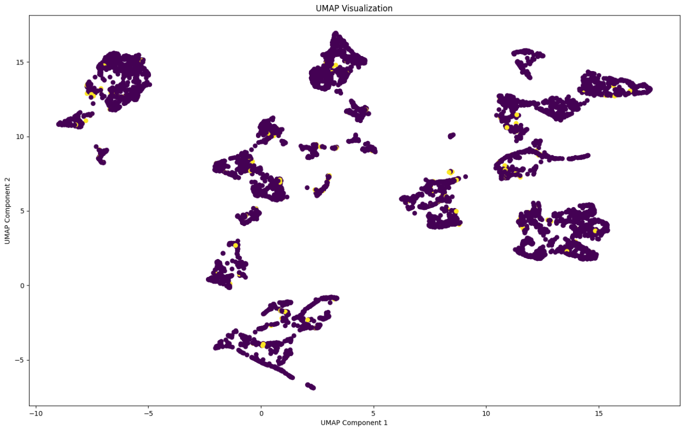
PCA Visualization

Principal Component Analysis (PCA) is applied to reduce the dimensionality of the dataset to two components. The resulting visualization demonstrates the distribution of instances in a 2D space:



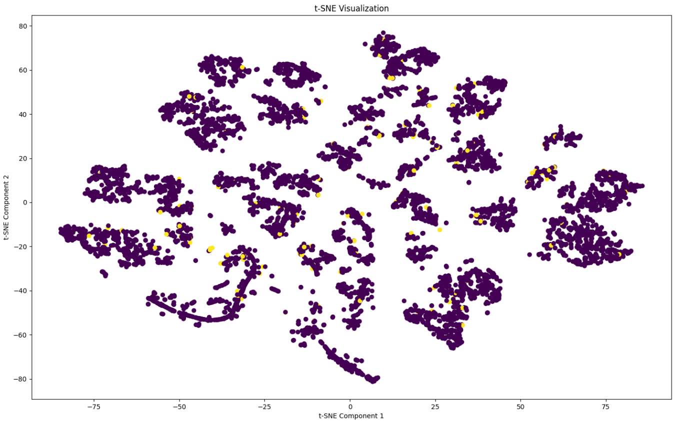
UMAP Visualization

Uniform Manifold Approximation and Projection (UMAP) is used for dimensionality reduction. The 2D visualization provides insights into the structure of the data:



t-SNE Visualization

t-Distributed Stochastic Neighbor Embedding (t-SNE) is employed to visualize the dataset in a 2D space, capturing local relationships between instances:



Observations

- The correlation matrix and feature impact analysis offer initial insights into potential relationships between features and the target variable.
- Dimensionality reduction techniques (PCA, UMAP, t-SNE) provide different perspectives on the distribution of instances, aiding in the exploration of the dataset's structure.

These visualizations serve as a foundation for further analysis and model building. Further investigation and experimentation can be conducted based on the patterns and insights gained from these exploratory data analyses.

3. Model Training

3.1 Logistic Regression (with Feature Selection)

3.1.1 Model Description

Logistic Regression is employed with feature selection using another logistic regression model. The selected features are used to train the final logistic regression model.

3.1.2 Result

Validation Accuracy: 89.32%

Cross-Validation Mean Accuracy: 89.31%

Test Accuracy: 89.32%

Precision, recall, and f1-score for both classes (0 and 1).

3.2 Random Forest

3.2.1 Model Description

A Random Forest classifier with 20 estimators is used for bankruptcy prediction.

3.2.2 Results

Validation Accuracy: 97.39%

Cross-Validation Mean Accuracy: 96.48%

Test Accuracy: 97.39%

Precision, recall, and f1-score for both classes.

3.3 XGBoost

3.3.1 Model Description

XGBoost, a powerful gradient boosting algorithm, is applied with specific configurations.

3.3.2 Results

Validation Accuracy: 97.88%
Cross-Validation Mean Accuracy: 97.44%
Test Accuracy: 97.88%
Precision, recall, and f1-score for both classes.

3.4 KNeighbourClassifier

3.4.1 Model Description

KNeighbourClassifier, a k-nearest neighbors algorithm, is introduced for bankruptcy prediction.

3.4.2 Results

Validation Accuracy: 93.75%
Cross-Validation Mean Accuracy: 93.41%
Test Accuracy: 93.75%
Precision, recall, and f1-score for both classes.

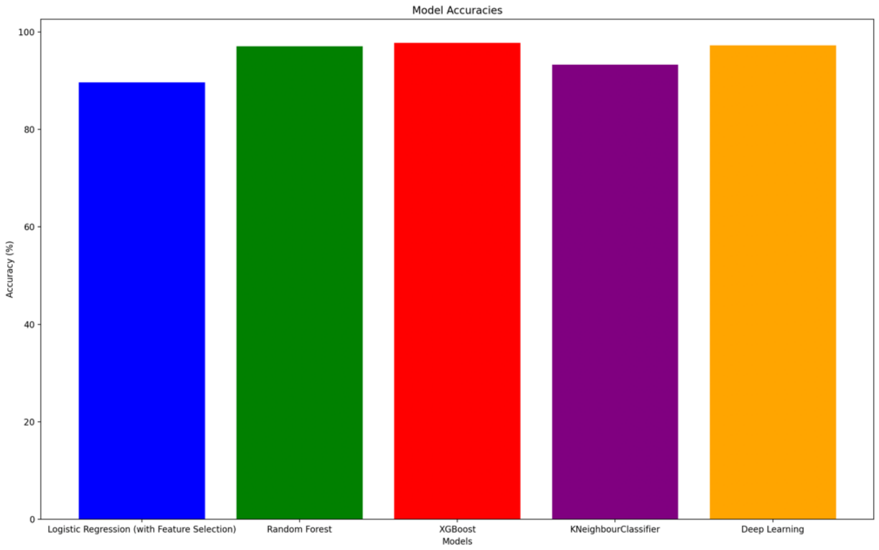
3.5 Deep Learning (MLPClassifier)

3.5.1 Model Description

A Multilayer Perceptron (MLP) Classifier is utilized for deep learning-based bankruptcy prediction.

3.5.2 Results

Validation Accuracy: 97.42%
Cross-Validation Mean Accuracy: 96.68%
Test Accuracy: 97.42%
Precision, recall, and f1-score for both classes.



Accuracy Plot of all models

4. Model Evaluation

4.1 Cross-Validation

Cross-validation is a pivotal element in assessing the robustness and generalizability of our predictive models. In this project, a common strategy known as k-fold cross-validation is employed. The dataset is divided into 'k' subsets (folds), and the model is trained and evaluated 'k' times, with each fold serving as the validation set exactly once.

The choice of 'k' (number of folds) influences the trade-off between computation time and result reliability. Here, a value of 'k=5' is selected, striking a balance between resource efficiency, and obtaining a representative estimate of model performance.

Cross-validation is particularly valuable in our context as it helps identify potential issues such as overfitting or underfitting. By training on different subsets of the data, we gain insights into how well the model generalizes to unseen data.

4.2 Performance Comparison

The models under consideration encompass Logistic Regression (with Feature Selection), Random Forest, XGBoost, KNeighbourClassifier, and Deep Learning. Each model is meticulously trained, validated, and tested, with their performances summarized below:

1. Logistic Regression (with Feature Selection):

- Strengths: This model combines the interpretability of logistic regression with feature selection, focusing on the most influential attributes.
- Weaknesses: Linear models may struggle with capturing complex, non-linear relationships in the data.

2. Random Forest:

- Strengths: Robust against overfitting, capable of handling non-linearity and interactions in the data.
- Weaknesses: Interpretability might be compromised due to the ensemble nature of the model.

3. XGBoost:

- Strengths: Excellent predictive performance, handles complex relationships, and incorporates feature importance.
- Weaknesses: Sensitive to overfitting; careful tuning of hyperparameters is crucial.

4. KNeighbourClassifier:

- Strengths: Simple and intuitive, effective for well-separated classes in feature space.
- Weaknesses: Performance may degrade with high-dimensional data, sensitive to outliers.

5. Deep Learning (MLPClassifier):

- Strengths: Powerful in capturing intricate patterns, suitable for complex tasks.
- Weaknesses: Prone to overfitting, requires careful tuning of architecture and hyperparameters.

The comparative analysis delves into the accuracy, precision, recall, and F1-score metrics for each model, providing a comprehensive understanding of their performance characteristics. This information guides the selection of the most suitable model for the task at hand.

5. Potential Improvements

5.1 Identified Pitfalls

During the training of the models, convergence warnings were observed for the Deep Learning model (MLPClassifier). These warnings indicate that the optimization algorithm did not converge within the maximum number of iterations specified. This convergence issue could affect the model's ability to learn and may impact predictive performance.

5.2 Proposed Improvements

To address the convergence warnings and enhance overall model performance, the following improvements are recommended:

Deep Learning Model (MLPClassifier):

- ***Increase Maximum Iterations:*** Allowing the optimization algorithm more iterations to converge might mitigate the convergence warning. The `max_iter` parameter in MLPClassifier can be adjusted accordingly.

Fine-Tune Hyperparameters:

Further hyperparameter tuning is recommended for the Deep Learning model. Parameters such as the learning rate, number of hidden layers, and neurons in each layer should be optimized for better convergence and performance.

General Recommendations:

- ***Ensemble Methods:*** Consider exploring ensemble methods, such as stacking or blending, to combine the strengths of multiple models and potentially improve overall predictive accuracy.

- ***Feature Engineering:*** Experiment with additional feature engineering techniques. Creating new meaningful features or transforming existing ones might provide models with more relevant information for decision-making.

5.3 Additional Experiments

To expand the scope of experimentation and potentially discover superior models, the following two additional experiments are proposed:

5.3.1. Algorithm Variation:

Experiment with alternative algorithms or variations of existing ones. For instance, consider different ensemble configurations, explore gradient boosting variants, or delve into other neural network architectures.

5.3.2. Feature Selection Methods:

Evaluate different feature selection methods beyond the Logistic Regression-based approach used in the current implementation. Techniques like recursive feature elimination or tree-based feature selection could be explored to identify the most informative attributes.

These additional experiments aim to provide a more comprehensive understanding of the dataset and the models' behavior, potentially leading to improved predictive performance.

6. Code and Dataset

Code repository : https://github.com/sundarmachani/CS-584-machine-Learning-Final-Project

Kaggle Dataset used : https://www.kaggle.com/datasets/fedesoriano/company-bankruptcy-prediction/data

7. Conclusion

This project delved into bankruptcy prediction, leveraging machine learning for valuable insights into company financial health. Key findings include:

1. Dataset Exploration:

- 'data.csv' offered a rich set of financial attributes.
- Preprocessing involved label encoding, feature scaling, and SMOTE for class imbalance.

2. Model Development:

- Explored algorithms: Logistic Regression, Random Forest, XGBoost, KNeighbourClassifier, and Deep Learning.
- Logistic Regression aided feature selection.

3. Model Evaluation:

- Robust cross-validation strategy used for evaluation.
- Comparative analysis highlighted algorithm strengths and weaknesses.

Final Thoughts:

This project lays a foundation for machine learning in bankruptcy prediction. Insights gained, coupled with recommendations, set the stage for model refinement. Continuous adaptation ensures these models stay effective in dynamically evolving financial landscapes.