Financial Distress Prediction using ML

A.Veerendra Nath,Sri varsha tandra

Decision Scientist, Bengaluru

Abstract:

The prediction of bankruptcy has become an area of growing interest for businesses seeking to mitigate financial losses due to unpaid debts. With the ability to store vast amounts of bankruptcy-related data, computers play a crucial role in making early and accurate predictions. Leveraging these datasets enables firms to assess financial risks proactively.

The dataset used in this study was sourced from the Taiwan Economic Journal, covering the period from 1999 to 2009. The definition of company bankruptcy. aligns.with the business regulations set forth by the Taiwan Stock Exchange.

Problem Statement:

The main objective of this project is to use various classification algorithms on bankruptcy dataset to predict bankruptcies with satisfying accuracies long before the actual event.

Dataset Information:

- Number of instances: 6819
- Number of attributes: 96

X5 - Realized Sales Gross Margin: Realize d Gross Profit/Net Sales

X6 - Operating Profit Rate: Operating Inco me/Net Sales

X7 - Pre-tax net Interest Rate: PreTax Income/Net Sales

X8 - After-
tax net Interest Rate: Net Income/Net Sale s

X9 - Nonindustry income and
expenditure/revenue:
Net Non-operating Income Ratio

X10 - Continuous interest rate (after tax):
Net Income-
Exclude Disposal Gain or Loss/Net Sales

X11 - Operating Expense Rate: Operating Expenses/Net Sales

Understanding Data:

Updated column names and description to make the data easier to understand (Y = O utput feature, X = Input features)

Y - Bankrupt? Class label 1 : Yes , 0: No

X1 - ROA(C) before interest and depreciat ion before interest: Return On Total Assets (C)

X2 - ROA(A) before interest and % after t ax: Return On Total Assets(A)

X3 - ROA(B) before interest and depreciat ion after tax: Return On Total Assets(B)

X4 - Operating Gross Margin: Gross Profit /Net Sales

X12 - Research and development expense rate: (Research and Development Expenses)/Net Sales

X13 - Cash flow rate: Cash Flow from Operating/Current Liabilities

X14 - Interestbearing debt interest rate: Interestbearing Debt/Equity

X15 - Tax rate (A): Effective Tax Rate

X16 - Net Value Per Share (B): Book Value Per Share(B)

X17 - Net Value Per Share (A): Book Value Per Share(A)

X18 - Net Value Per Share (C): Book Value Per Share(C)

X19 - Persistent EPS in the Last Four Seasons: EPS-Net Income

X20 - Cash Flow Per Share

X21 - Revenue Per Share (Yuan ¥): Sales Per Share

X22 - Operating Profit Per Share (Yuan ¥): Operating Income Per Share

X23 - Per Share Net profit before tax (Yuan ¥): Pretax Income Per Share

X24 - Realized Sales Gross Profit Growth Rate

X25 - Operating Profit Growth Rate: Operating Income Growth

X26 - Aftertax Net Profit Growth Rate: Net Income Growth

X27 - Regular Net Profit Growth Rate: Continuing Operating Income after Tax Growth

X28 - Continuous Net Profit Growth Rate: Net Income-
Excluding Disposal Gain or Loss Growth

X29 - Total Asset Growth Rate: Total Asset Growth

X30 - Net Value Growth Rate: Total Equity Growth

X31 - Total Asset Return Growth Rate Ratio: Return on Total Asset Growth

X32 - Cash Reinvestment %: Cash Reinvestment Ratio

X33 - Current Ratio

X34 - Quick Ratio: Acid Test

X35 - Interest Expense Ratio: Interest Expenses/Total Revenue

X36 - Total debt/Total net worth: Total Liability/Equity Ratio

X37 - Debt ratio %: Liability/Total Assets

X38 - Net worth/Assets: Equity/Total Assets

X39 - Longterm fund suitability ratio (A): (Longterm Liability+Equity)/Fixed Assets

X40 - Borrowing dependency: Cost of Interest-bearing Debt

X41 - Contingent liabilities/Net worth: Co
ntingent Liability/Equity

X42 - Operating profit/Paidin capital: Operating Income/Capital

X43 - Net profit before tax/Paidin capital: Pretax Income/Capital

X44 - Inventory and accounts receivable/Net value: (Inventory+Accounts Receivables)/Equity

X45 - Total Asset Turnover

X46 - Accounts Receivable Turnover

X47 - Average Collection Days: Days Receivable Outstanding

X48 - Inventory Turnover Rate (times)

X49 - Fixed Assets Turnover Frequency

X50 - Net Worth Turnover Rate (times): Equity Turnover

X51 - Revenue per person: Sales Per Employee

X52 - Operating profit per person: Operation Income Per Employee

X53 - Allocation rate per person: Fixed Assets Per Employee

X54 - Working Capital to Total Assets

X55 - Quick Assets/Total Assets

X56 - Current Assets/Total Assets

X57 - Cash/Total Assets

X58 - Quick Assets/Current Liability

X59 - Cash/Current Liability

X60 - Current Liability to Assets

X61 - Operating Funds to Liability

X62 - Inventory/Working Capital

X63 - Inventory/Current Liability

X64 - Current Liabilities/Liability

X65 - Working Capital/Equity

X66 - Current Liabilities/Equity

X67 - Longterm Liability to Current Assets

X68 - Retained Earnings to Total Assets

X69 - Total income/Total expense

X70 - Total expense/Assets

X71 - Current Asset Turnover Rate: Current Assets to Sales

X72 - Quick Asset Turnover Rate: Quick Assets to Sales

X73 - Working capitcal Turnover Rate: Working Capital to Sales

X74 - Cash Turnover Rate: Cash to Sales

X75 - Cash Flow to Sales

X76 - Fixed Assets to Assets

X77 - Current Liability to Liability

X78 - Current Liability to Equity

X79 - Equity to Long-term Liability

X80 - Cash Flow to Total Assets

X81 - Cash Flow to Liability

X82 - CFO to Assets

X83 - Cash Flow to Equity

X84 - Current Liability to Current Assets

X85 - Liability-
Assets Flag: 1 if Total Liability exceeds T otal Assets, 0 otherwise

X86 - Net Income to Total Assets

X87 - Total assets to GNP price

X88 - No-credit Interval

X89 - Gross Profit to Sales

X90 - Net Income to Stockholder's Equity

X91 - Liability to Equity

X92 - Degree of Financial Leverage (DFL)

X93 - Interest Coverage Ratio (Interest ex pense to EBIT)

X94 - Net Income Flag: 1 if Net Income is Negative for the last two years, 0 otherwis e

X95 - Equity to Liability

○ Steps Involved:

- **Data Collection**

To begin addressing the problem, we will first load our dataset, which is provided in **.csv** format, into a dataframe. This involves mounting the drive and importing the **CSV** file into a dataframe for further analysis.

- **Splitting the Data in two categories:**

The records are observed to be highly imbalanced. Thus it is necessary to consider balancing the dataset through "Upsampling or Downsampling" techniques.

Through df.info(), we observed that we have a majority of "float64" data. The categorical data is distinguished as binary 1 and 0, thus stored as "int64". We separate the numeric and categoric data to analyze our dataset.

- **Exploratory Data Analysis**

Once the dataset was loaded, I checked for duplicate and null values, but none were found.The dataset lacks predictive power for linear algorithms, indicating that traditional linear models may not effectively capture patterns in the data. As a result, linear modeling techniques are not suitable for this dataset. Alternative approaches, such as non-linear models (e.g., Random Forest, Gradient Boosting, Neural Networks), should be considered for better predictive performance.

- **Data Modelling**

The dataset is highly imbalanced. Thus, be fore training the model, we need to deal wi th this data. For this we need to take following steps:

- Split the dataset into training and testi ng sets (80% - 20%). We preserve the 20% testing set for the final evaluation
- Through "Stratified K Fold CrossVali dation" we will now distribute the 80

% training set into further training and testing splits.

- Since we are dealing with over 50 features, we use "Randomized Search Cro ss Validation" as this technique proves to perform better with many features.

Fitting different Models

For Model fitting, I tried various classification algorithms like:

Logistic Regression

K-Nearest Neighbors (KNN)

Decision Tree

Support Vector Machine (Linear Kernel)

SVM(RBF Kernel)

Neural Network

Random Forest

Gradient Boosting

Feature Selection I Have DRT for feature selection.

Algorithms:

1. Random Forest Classifier:

The Random Forest, also known as the Random Decision Forest, is a supervised machine learning algorithm widely used for classification, regression, and various predictive tasks. This model constructs multiple decision trees, each trained on a randomly selected subset of the dataset. Instead of relying on a single tree, the algorithm aggregates predictions from all trees by taking a majority vote (for classification) or averaging the outputs (for regression) to enhance accuracy and reduce overfitting.

Upon implementing this model, we obtained the following performance scores:

Model Score	Precision	Recall	F1 score	support
95.51%	0.81	0.92	0.84	893

2. K Nearest Neighbor

K-nearest neighbors (KNN) is a type of supervised learning algorithm used for both regression and classification. KNN tries to predict the correct class for the test data by calculating the distance between the test data and all the training points. Then select the K number of points which is closest to the test data. The

KNN algorithm calculates the probability of the test data belonging to the classes of 'K' training data and class which holds the highest probability will be selected. When we implement this model, we get the following scores:

Model Score	Precision	Recall	F1 score	support score
94.98%	1.00	0.98	0.99	846

3. Support Vector Classifier

The objective of Support Vector Classifier algorithm is to find a hyperplane in an N-dimensional space that distinctly classifies the data points. The dimension of the hyperplane

depends upon the number of features. If the number of input features is two, then the hyperplane is just a line. If the number of input features is three, then the hyperplane becomes a 2-D plane. It becomes difficult to imagine when the number of features exceeds three. When we implement this model, we get the following scores:

Model Score	Precision	Recall	F1 score	ROC-AUC score
94.98%	0.390000	0.870000	0.540000	0.910000

4. Logistic Regression:

Logistic regression is basically a supervised classification algorithm. In a classification problem, the target variable (or output), y, can take only discrete values for a given set of features (or inputs), X. Logistic regression becomes a classification technique only when a decision threshold is brought into the picture. The setting of the threshold value is a very important aspect of Logistic regression and is dependent on the classification problem itself. When we implement this model, we get the following scores:

Model Score	Precision	Recall	F1 score	ROC-AUC score
86.53%	0.19	0.87	0.31	0.87

5. Decision Tree Classifier

Decision Tree is a Supervised Machine Learning Algorithm that uses a set of rules to make decisions, similarly to how humans make decisions. One way to think of a Machine Learning classification algorithm is that it is built to make decisions. The intuition behind Decision Trees is that you use the dataset features to create **yes/no**
questions and continually split the dataset until you isolate all data points belonging to each class. With this process we are organizing the data in a tree structure. When we implement this model, we get the following scores:

Model Score	Precision	Recall	F1 score	ROC-AUC score
84.68%	0.18	0.90	0.31	0.87

- **Applying PCA for Feature Extraction:**

  The Scikit-learn API provides the PCA (Principal Component Analysis) class for dimensionality reduction, allowing us to transform a high-dimensional dataset into a smaller set of principal components while retaining maximum variance. PCA helps in eliminating less significant features, reducing data complexity, and improving training efficiency. This process is particularly useful when working with large datasets, as it minimizes redundancy and speeds up model training. I applied PCA to the Random Forest Classifier and K-Nearest Neighbor Classifier, but the results were not satisfactory.

Challenges Faced during the Project

Balancing the trade-off between recall and precision posed a significant challenge. Selecting the right features and techniques required careful consideration. Exploring literature and resources to understand the problem and identify solutions was an intensive process. Meeting deadlines felt demanding at times, but in the end, everything came together successfully.

Conclusion:

A significant portion of firms in the dataset have experienced continuous losses over the past two years, as reflected by their negative net income. However, only a small fraction of these companies have actually declared bankruptcy.

Key financial attributes, such as **Debt Ratio %, Current Liability to Assets, and Current Liability to Current Assets**, exhibit a strong positive correlation with bankruptcy, suggesting that higher values of these indicators increase financial vulnerability.

Machine learning models effectively predict corporate bankruptcy, with SVM (RBF kernel) achieving the highest accuracy (95.43%). Key financial attributes strongly correlate with bankruptcy risk, highlighting the importance of data-driven early warning systems.

References:

scikit-learn.org, 'RandomForestClassifier'. [Online].

Available: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html/

Nima Beheshti, 'Random Forest Classification'. [Online].

Available: https://towardsdatascience.com/random-forest-classification-678e551462f5/

en.wikipedia.org, 'k-nearest neighbors algorithm'. [Online].

Available: https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm/

Sai Patwardhan, 'Simple understanding and implementation of KNN algorithm!'. [Online].

Available: https://www.analyticsvidhya.com/blog/2021/04/simple-understanding-and-implementation-of-knn-algorithm//

Youtube.com, 'K nearest neighbors classification with python code'. [Online].

Available: https://www.youtube.com/watch?v=CQveSaMyEwM&ab_channel=codebasics/

Youtube.com, 'Introduction to Random Forest'. [Online].

Available: https://www.youtube.com/watch?v=F9uESCHGjhA&ab_channel=CampusX/

Pellegrino, M., Lombardo, G., Adosoglou, G., Cagnoni, S., Pardalos, P. M., & Poggi, A. (2024). A Multi-Head LSTM Architecture for Bankruptcy Prediction with Time Series Accounting Data. Future Internet, 16(3), 79.

Machine Learning for Bankruptcy Prediction in the American Stock Market: Dataset and Benchmarks - Future Internet MDPI 2022