# Company Bankruptcy Prediction (Kaggle)

Ritesh Kumar

2024WI\_MS\_DSP\_422-DL\_SEC61: Practical Machine Learning

Module 4 Assignment

# Company Bankruptcy Prediction

# Donald Wedding and Narayana Darapaneni

# January 29, 2024

**Approach**

We start with a comprehensive Exploratory Data Analysis (EDA), commencing with the extraction of descriptive statistics for the 'Bankrupt?' column, serving as the target variable for prediction. We also summarize the dataset by computing the count, mean, standard deviation, minimum, quartiles, and maximum values, for all the 96 columns.

A graph with a bar and a number of numbers

Description automatically generated with medium confidence

A graph of a graph

Description automatically generatedA graph with numbers and lines

Description automatically generated with medium confidence

There were no missing values or duplicate rows in the dataset.

Outlier removal in the independent variables for a dataset with a highly imbalanced dependent variable like this one (where bankrupt cases are the minority) could potentially eliminate valuable information. Since bankruptcies are rare events, the characteristics that lead to bankruptcy may be present as outliers in the independent variables. These "outliers" might be critical in predicting the rare event of bankruptcy. If they were removed, the model's ability to generalize and identify the risk of bankruptcy could be significantly impaired. Therefore, we decided to not identify or remove the outliers

Next, we identified the top-20 features that have the highest correlation with Bankrupt, followed it up by plotting a heatmap depicting the strength of these 20 features amongst themselves and with ‘Bankrupt?’. To avoid multi-collinearity, we removed 20 variables that had a significantly high corelation (>0.95) amongst themselves.

We plotted the correlation heatmap of ‘Bankrupt?’ and the new top-20 highly correlated features, and boxplots and distribution-plots of these 20 features.

The dataset was split into train and test, in 80:20 ratio, and the data was scaled using StandardScaler.

Our dataset showcases a significant class imbalance with a vast majority of cases being non-bankrupt (6599) and a small minority being bankrupt (220). In such scenarios, logistic regression models tend to be biased towards the majority class, leading to poor classification performance on the minority class. SMOTE (Synthetic Minority Over-sampling Technique) generates synthetic samples for the minority class, helping to balance the dataset. This balance allows the logistic regression model to learn a more generalized decision boundary, improving its ability to correctly identify cases of bankruptcy, which is critical for the model's predictive performance. By enhancing the representation of the minority class, SMOTE helps in improving the sensitivity (recall) and precision of the model, ensuring that both classes are predicted more accurately, rather than the model overwhelmingly predicting the majority class. We use SMOTE.

We start modelling by deploying the **Logistic Regression Model** and trying different hyperparameters in the model. The best model returns:

* Accuracy: 0.8673
* Precision: 0.1875
* Recall: 0.7647
* F1 Score: 0.3012

A graph with a line

Description automatically generatedA graph of a positive rate

Description automatically generated with medium confidence

The ROC curve shows an area under the curve (AUC) of 0.82, indicating a good ability of the model to distinguish between the positive and negative classes. This is a significant improvement over random chance (AUC = 0.50).

The Precision-Recall chart, however, presents an AUC of 0.48, which isn't far from random for precision-recall performance but is typical in the context of imbalanced datasets.

The hyperparameters {'C': 800, 'penalty': 'l1'} indicate that the model is using L1 regularization with a relatively high penalty strength, which tends to produce a model with more feature selection (due to the L1 norm's tendency to push coefficients to exactly zero).

The accuracy of 0.8621 is relatively high, but accuracy is not a reliable metric in the context of imbalanced classes. The precision of 0.1814 is low, indicating that when the model predicts bankruptcy, it is correct only about 18% of the time. The recall of 0.7647 is high, which means the model is able to identify approximately 76% of all actual bankrupt cases. This suggests that the model is biased towards predicting the minority class, which is often desirable in scenarios where the cost of missing a positive case is high (such as predicting bankruptcy).

The F1 score, which balances precision and recall, is 0.2932, reflecting a moderate trade-off between precision and recall. This score is not high, indicating that there is still room for improvement in achieving a balance between precision and recall.

In summary, the model is a significant step in the right direction, especially in terms of recall and ROC AUC, but the low precision and moderate F1 score indicate that the model may still be improved, perhaps by further adjusting the class balance or the regularization strength.

The **Gaussian Naïve Bayes Model** returns:

* Accuracy: 0.81473
* Precision: 0.0389
* Recall: 0.9215
* F1 Score: 0.0747

A graph of a positive rate

Description automatically generated with medium confidence A graph with a line

Description automatically generated

The Receiver Operating Characteristic (ROC) curve shows an area under the curve (AUC) of 0.52, which suggests that the model's ability to distinguish between positive and negative classes is barely better than random chance, which would have an AUC of 0.50. This indicates that the model is not effective at correctly classifying the positive class.

The Precision-Recall (PR) chart presents an AUC of 0.48, which indicates that the model performs poorly in terms of both precision and recall, as this value is even less than random chance for PR performance. However, it should be noted that such outcomes can be characteristic of datasets where the positive class is very rare.

Despite the poor AUC scores, the model exhibits a surprisingly high recall of 0.9216, meaning it correctly identifies approximately 92% of all actual positive cases. This is an indication that the model is quite sensitive to the positive class, but this often comes at the expense of precision.

The model's precision is extremely low at 0.039, which implies that when the model predicts an instance as positive, it is correct only about 4% of the time. This could lead to a large number of false positives, which might be costly or undesirable depending on the application.

An accuracy of 0.1474 is quite low, which highlights that the model is incorrect in its predictions most of the time. This is not uncommon in imbalanced datasets where the metric can be misleading.

The F1 Score for the Gaussian Naive Bayes model is 0.0748, which is a harmonic mean of precision and recall. This low score reflects the imbalance between the model's high recall and very low precision, indicating poor overall performance.

The **Support Vector Machine Model** returns:

* Accuracy: 0.9618
* Precision: 0.4545
* Recall: 0.0981
* F1 Score: 0.1612

A graph of a positive rate

Description automatically generated with medium confidenceA graph with a line

Description automatically generated

The Receiver Operating Characteristic (ROC) curve for this model displays an area under the curve (AUC) of 0.55, which suggests only a slight improvement over random guessing, which would have an AUC of 0.50. This indicates that the model's discriminative ability to correctly classify the positive cases is marginally better than chance.

The Precision-Recall (PR) chart shows an even lower AUC of 0.29, reflecting that the model is particularly weak in terms of precision and recall. This is further emphasized by the model's precision of 0.4545, indicating that when the model predicts a positive outcome, it is correct less than half of the time.

Moreover, the recall of the model is 0.0980, which means it identifies less than 10% of all actual positive cases. This low recall suggests that the model is not sensitive enough to the positive class, missing many positive instances. Despite these limitations, the model achieves an accuracy of approximately 0.962, which could be misleading as it does not reflect the model's poor performance in correctly classifying the positive class.

Lastly, the F1 Score, which balances precision and recall, is very low at 0.1613. This score is consistent with the poor AUC values and reflects the model's inadequate performance in classifying the positive class correctly.

**Management Recommendations:** To compare the three models, we plotted the confusion matrices:

A blue and white squares with black text

Description automatically generated

Logistic Regression demonstrates a reasonable trade-off between precision and recall, with an accuracy of approximately 86.7%, precision at 18.8%, and a recall of 76.5%, leading to an F1 score of around 0.300. Naive Bayes offers high recall at 92.2%, indicating its strength in identifying actual bankruptcy cases, but it suffers from a higher false positive rate, resulting in a precision of 23.4% and an overall F1 score of 0.372. SVM, on the other hand, achieves high accuracy at 96.2% but is heavily biased towards predicting the non-bankruptcy class, evidenced by a low recall of 9.8%. This bias yields a precision of 45.5% and a notably lower F1 score of 0.161. While SVM may appear superior in terms of accuracy, its practical usefulness is questionable due to its poor recall. Conversely, the Naive Bayes model, with the highest F1 score, suggests a more balanced performance, especially important in scenarios where failing to detect actual bankruptcies could have significant consequences. Hence, despite having the lowest accuracy, Naive Bayes could be considered the most effective model for predicting bankruptcy.

The submission results can be viewed at <https://www.kaggle.com/riteshrk/classification-lr-nb-svm/edit>.

**Code**