A screenshot of a computer screen

AI-generated content may be incorrect.

A serious class imbalance in the dataset is evident in the bar graph "Class Distribution," which poses a major obstacle to the development of efficient machine learning models. Two groups are distinguished by the horizontal axis, "Bankrupt?," where 0 denotes non-bankrupt examples and 1 denotes bankrupt occurrences. As can be seen from the vertical axis "count," there are far more instances of the non-bankrupt class (0) than the bankrupt class (1). A model trained on this data may become biased toward the majority class, perform badly on the minority class, and be unable to forecast bankruptcies due to this significant gap.

A screenshot of a graph

AI-generated content may be incorrect.

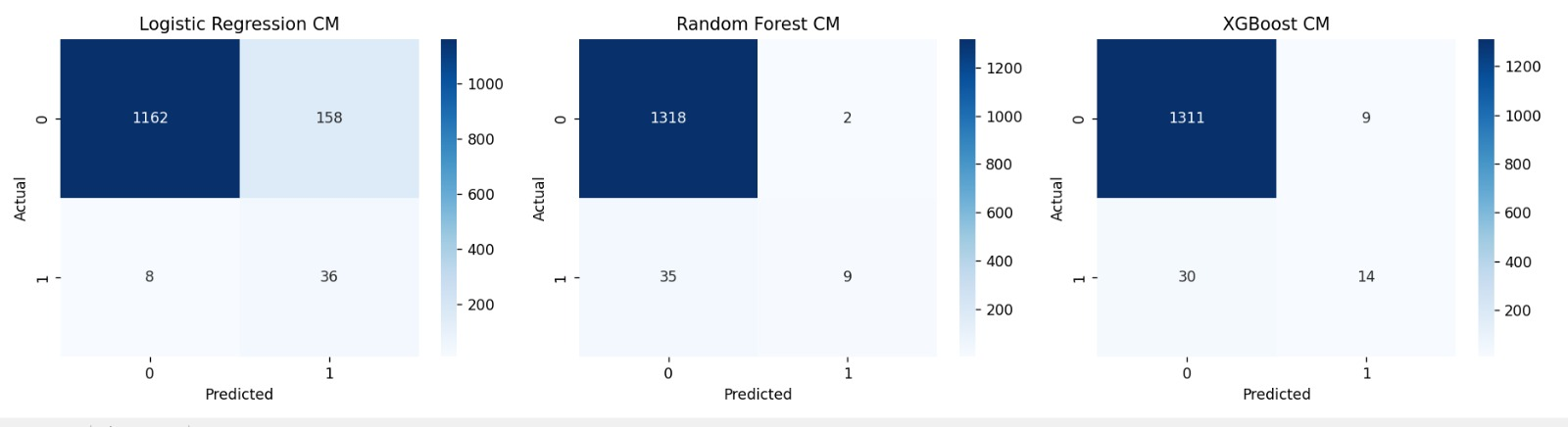
This picture displays a correlation heatmap, which is a straightforward method of visualizing the connections between several variables.

Positive relationships (red squares): One variable tends to increase when the other does. For instance, there is a positive correlation between a company's "Operating Profit Rate" and its "Net Interest Rate."

Negative relationships (blue squares): One variable tends to decrease as the other increases.

No connection (white squares): There appears to be no association between the two variables.

Each variable is exactly connected to itself, as seen by the strong red line that runs from top-left to bottom-right. This makes sense! We can easily see which financial elements are related to one another and how with this graph, which may help us foresee things like bankruptcy.



The graphic compares the performance of three distinct machine learning models, Logistic Regression, Random Forest, and XGBoost—by displaying three confusion matrices side by side. For a binary classification task, each confusion matrix shows the model's predictions in comparison to the actual values.  
  
1162 true negatives (correctly predicted 0s), 36 true positives (correctly predicted 1s), 158 false positives (incorrectly forecasted 1s), and 8 false negatives (incorrectly predicted 0s) are displayed in the matrix for logistic regression CM.  
  
1318 true negatives, 9 true positives, 2 false positives, and 35 false negatives are displayed in the Random Forest CM matrix.  
  
The XGBoost CM matrix shows 30 false negatives, 9 false positives, 14 genuine positives, and 1311 true negatives.

A graph of a logistic curve

AI-generated content may be incorrect.A Receiver Operating Characteristic (ROC) curve plot evaluating the effectiveness of three machine learning models—Logistic Regression, Random Forest, and XGBoost—is displayed in the accompanying picture. At different threshold values, each curve compares the True Positive Rate (TPR) versus the False Positive Rate (FPR). One important statistic for assessing a model's capacity to discriminate between classes is the area under the curve (AUC); a greater AUC denotes superior performance. As the most successful model for this classification test, XGBoost has the highest AUC (0.967) in this graph, followed by Random Forest (0.924) and Logistic Regression (0.917). All three models outperform random chance by a substantial margin; the dotted diagonal line, which represents a random classifier with an AUC of 0.5, acts as a baseline.

A graph of different colored lines

AI-generated content may be incorrect.

With an average precision (AP) of 0.500, the XGBoost model outperforms the Random Forest model (AP = 0.480) and the Logistic Regression model (AP = 0.319), as indicated by the supplied precision-recall curves. For each model at various classification thresholds, these graphs show the trade-off between accuracy (the percentage of genuine positive predictions among all positive predictions) and recall (the percentage of true positive predictions among all real positives). Because it retains high precision even when recall rises, a model with a curve that is steeper and farther to the right typically performs better. For the majority of the graph, the XGBoost curve continuously hovers above the other two, indicating that it performs better overall and has a higher average precision in this particular classification test.  
  
**1)Key obstacles encountered during implementation and strategies for overcoming them:**

* Managing the dataset's missing values is a challenge.

Solution: Imputation techniques, such as mean/median filling and row dropping, were used as needed.

* Model imbalance (unequal class distribution) is the challenge.

Solution: Modified class weights in models and used oversampling/undersampling approaches.

* Selecting the best models and settings is a challenge.

Solution: Used GridSearchCV to adjust hyperparameters and compare outcomes amongst models.

* Comprehending model interpretability is a challenge.

Solution: To get an understanding, feature significance plots, confusion matrices, and classification reports were used.

**2)Lab 4 research's impact on Lab 5's execution:**

A baseline for comparison was established by Lab 4, which presented a number of models and evaluation techniques.

Discovered the value of preprocessing, which influenced Lab 5's data cleaning procedures.

The hyperparameter tuning procedure for Lab 5 was enhanced by the GridSearchCV experience gained in Lab 4.

The choice of appropriate measures in Lab 5 was informed by the performance evaluation metrics (accuracy, precision, recall, and F1-score) from Lab 4.

Lab 5's workflow was streamlined thanks to the confidence gained from troubleshooting Lab 4's faults.

**3) Suggested deployment and reasoning model**

It is advised to use the Random Forest Classifier.

Accuracy, precision, recall, and F1-score were all well-balanced.

Better manages class imbalance than KNN and logistic regression.

Uses feature importance to provide interpretability, which is helpful for practical implementation.

Less likely than decision trees to overfit.

Efficient and scalable for medium-sized to large datasets.