Introduction

Predicting economic recessions has become extremely valuable due to the robust nature of global economics. Traditional economic forecasting has relied on classical statistical models and human expertise, leading to conflicting and mixed results. We believe that with machine learning, we can enhance the accuracy of recession predictions by analyzing historical economic data. We hope to not only be able to predict when a recession is likely to occur, but also identify how certain metrics contribute to the likelihood of a recession.

Problem Definition

Our objective is to construct an ML model that can predict the onset of a recession using key economic metrics. Since recessions are influenced by multiple factors such as economic metrics (GDP, unemployment rate, inflation) and market data (stock prices, bond yields), machine learning models can process and analyze these high dimensional relationships more effectively than traditional models.

Methods

Data Preprocessing and Feature Selection

To build an effective machine learning model, meticulous data preprocessing and feature selection is crucial since our dataset has lots of metrics and attributes. The dataset used for this project includes various economic indicators and a recession indicator to denote periods of economic downturns in the United States. The recession periods identified in the dataset are as follows:

- 1945: Post-WWII economic adjustment
- 1949: Economic slowdown after wartime economy shift

- 1953: End of the Korean War, leading to an economic downturn
- 1957–1958: Recession due to monetary tightening and declining industrial production
- 1960–1961: Recession marked by slowdowns in manufacturing and other industries
- 1969–1970: Recession primarily due to high inflation and tightening monetary policy
- 1973–1975: Oil crisis, high inflation, and stock market crash
- 1980: Economic downturn due to monetary tightening to combat inflation
- 1981–1982: Another deep recession to curb inflation with high interest rates
- 1990–1991: Recession following the savings and loan crisis and Gulf War
- 2001: Dot-com bubble burst and economic slowdown post-9/11
- 2007–2009: Great Recession, triggered by the financial crisis and housing market collapse
- 2020: COVID-19 pandemic recession due to lockdowns and economic shutdowns

The data is first processed using the Pandas library to import the dataset. We selected 21 key economic indicators that are believed to influence recessions. These features include the following:

- Broad money growth (annual %)
- Claims on private sector (annual growth as % of broad money)
- Consumer price index (2010 = 100)
- Domestic credit to private sector (% of GDP)
- GNI growth (annual %)
- General government final consumption expenditure (% of GDP)
- Gross capital formation (% of GDP)
- Imports of goods and services (annual % growth)
- Interest payments (% of expense)
- Labor force participation rate for ages 15-24, male (%) (national estimate)
- Lending interest rate (%)
- Machinery and transport equipment (% of value added in manufacturing)
- Merchandise trade (% of GDP)
- Net barter terms of trade index (2015 = 100)

- Net lending (+) / net borrowing (-) (% of GDP)
- Oil rents (% of GDP)
- Real interest rate (%)
- Risk premium on lending (lending rate minus treasury bill rate, %)
- Trade (% of GDP)
- Unemployment, youth male (% of male labor force ages 15-24) (national estimate)
- Unemployment, youth total (% of total labor force ages 15-24) (national estimate)

Missing values in the dataset were imputed using the mean strategy to ensure that the model is not skewed by incomplete data. Additionally, the features are standardized using standard scalar (provided by the Scikit-learn library) to ensure that each feature contributes equally to the model's performance.

Principal Component Analysis (PCA)

Feature engineering via Principal Component Analysis (PCA) was undergone on the dataset to further understand the underlying structure of the economic indicators and reduce dimensions, highlighting key features. Components that capture the most variance in our data were identified to facilitate better visualization and improve model performance. Two principal components were used to plot and understand the data in a two-dimensional space. We underwent the following steps in the PCA process:

1. PCA was initialized with two components and fitted to our standardized dataset.
2. The explained variance ratio and singular values were examined to understand how much variance is captured by each principal component, as well as understand the scaling margin of each principal component.

3. The component loadings were analyzed to identify which features contribute most to each principal component.
4. The data points were plotted in the space of the first two principal components, distinguishing between recession and non-recession periods. The component loadings were also visualized via a heatmap.

Model Development and Evaluation

A logistic regression model was employed via the Scikit-learn library for model development, training, and evaluation, since it serves as a robust binary classifier that can identify recessions (1) or non-recessions (0). The approach is outlined as follows:

1. The dataset was split 80-20 for both training and testing. Features were standardized using ‘StandardScalar’ to ensure optimal model performance.
2. A logistic regression model was trained on both the raw scaled data and PCA-transformed data to establish baseline and enhanced performance metrics.
3. To understand the classifier's decision boundary, we reduced the data to two principal components and visualized the model’s classification of points in the subspace.
4. We performed cross-validation using Stratified K-Fold to ensure robustness and generalizability. Evaluation metrics included accuracy, confusion matrix, classification report, and ROC AUC score.

Afterwards, we implemented K-Nearest Neighbors (KNN) as another approach for recession prediction for its versatility and robustness on non-linear data. We undertook the following steps:

1. Standardized the dataset to ensure all features contribute equally. PCA was employed again to reduce the dimensionality of the dataset to two components.
2. Since recession periods are inherently rarer than non-recession periods, we addressed class imbalance via the following techniques, which were then later combined into a pipeline to transform the training data:
   - SMOTE to generate more data points for recession periods.
   - Random Undersampling to reduce samples from the majority class.
3. We used GridSearchCV with k values ranging from 1 to 30 to find the optimal number of neighbors that maximized cross-validation accuracy.
   - k = 20 (best value) was selected for final testing.
4. The KNN model was trained on the resampled dataset using uniform weights. Standard metrics from the logistic regression model were reapplied.
5. We utilized two principal components from PCA to plot the decision regions for the trained KNN model, helping us visualize how the model classified recession and non-recession periods.

Lastly, we deployed a Random Forest model on the dataset due to its ability to find complex, non-linear relationships between features and our binary classifiers, as well as providing a visual representation of each feature's importance. We underwent the following procedures:

1. Standardized the dataset to ensure all features contribute equally. Applied PCA to reduce data to two dimensions for visualization.

2. Trained two separate Random Forest classifiers: One on the raw standardized features for feature importance analysis and another on the PCA-transformed data. The number of decision trees (n) was initially set to 100 and later tuned.
3. GridSearchCV was utilized again to optimize our n hyperparameter, with values ranging from 1 to 200, plotted against cross-validation accuracy. The number of trees was selected based on the maximized cross-validation accuracy.
4. A plot for the decision boundaries for the PCA-transformed dataset was generated to analyze the respective classifying recession and non-recession regions.
5. Feature importance was visualized using a horizontal bar plot, which was later sorted to identify the most significant recession predictor.
6. Model performance was evaluated using the same metrics earlier (test accuracy, confusion matrix, and classification report)

Results and Discussion

Principal Component Analysis (PCA)

Explained Variance and Singular Values:

- Explained Variance Ratio:
  - PC1: 33.68% variance data explained
  - PC2: 18.62% variance data explained
- Singular Values:
  - PC1: 21.275
  - PC2: 15.820

To visualize the distribution of recession and non-recession periods in the space defined by the first two principal components, we created scatter plots with a finalized version illustrated in figure 1. Recession periods are highlighted in red, while non-recession periods are shown in blue.

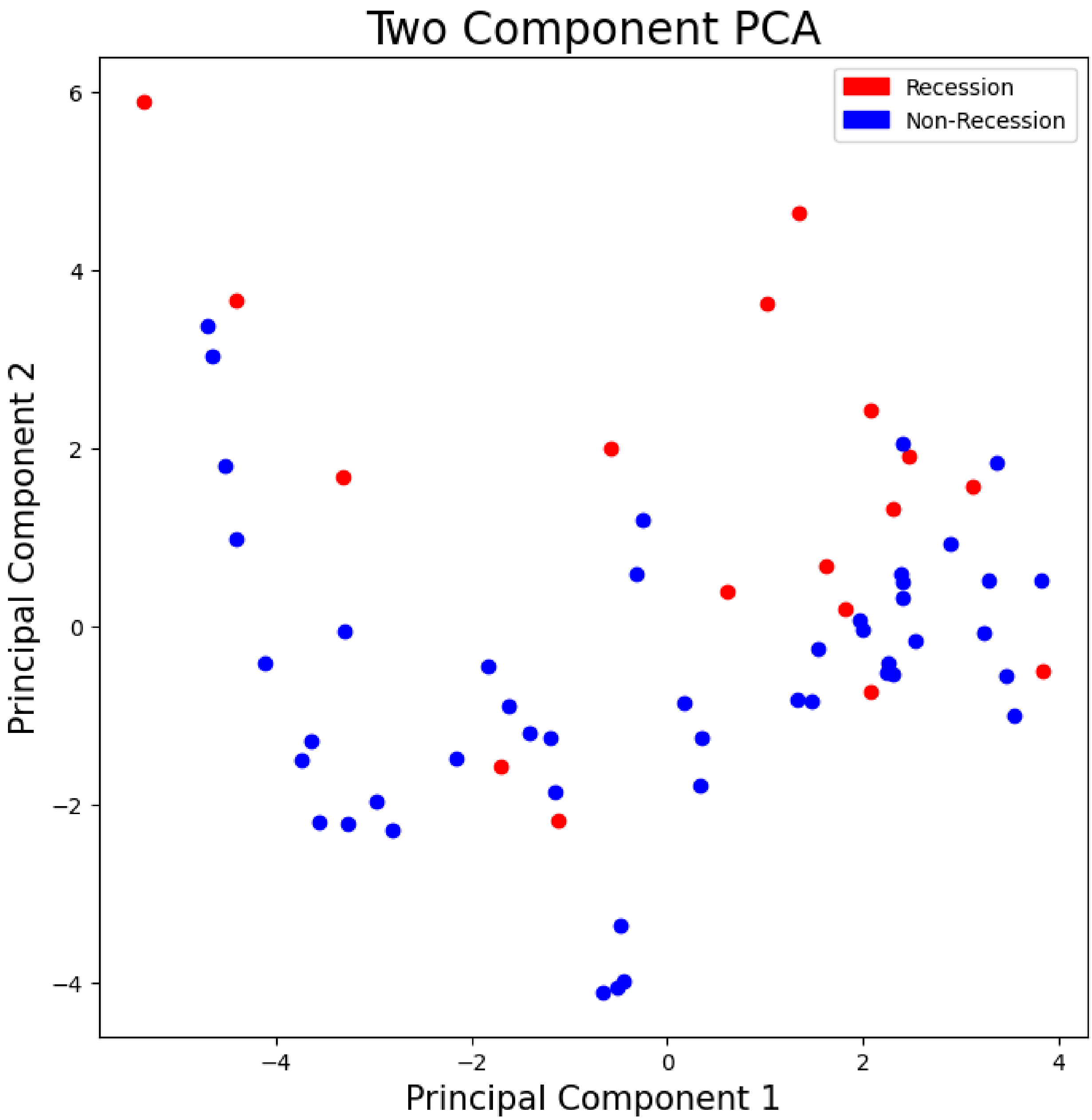

Figure 1: Scatter Plot of Two Principal Components (from PCA section)

Component Loadings:

- Top Features for PC1:
    1. Labor force participation rate for ages 15-24, male (%) (national estimate)
    2. Consumer price index (2010 = 100)
    3. Domestic credit to private sector (% of GDP)
- Top Features for PC2:

1. Unemployment, youth total (% of total labor force ages 15-24) (national estimate)
2. General government final consumption expenditure (% of GDP)
3. Unemployment, youth male (% of male labor force ages 15-24) (national estimate)

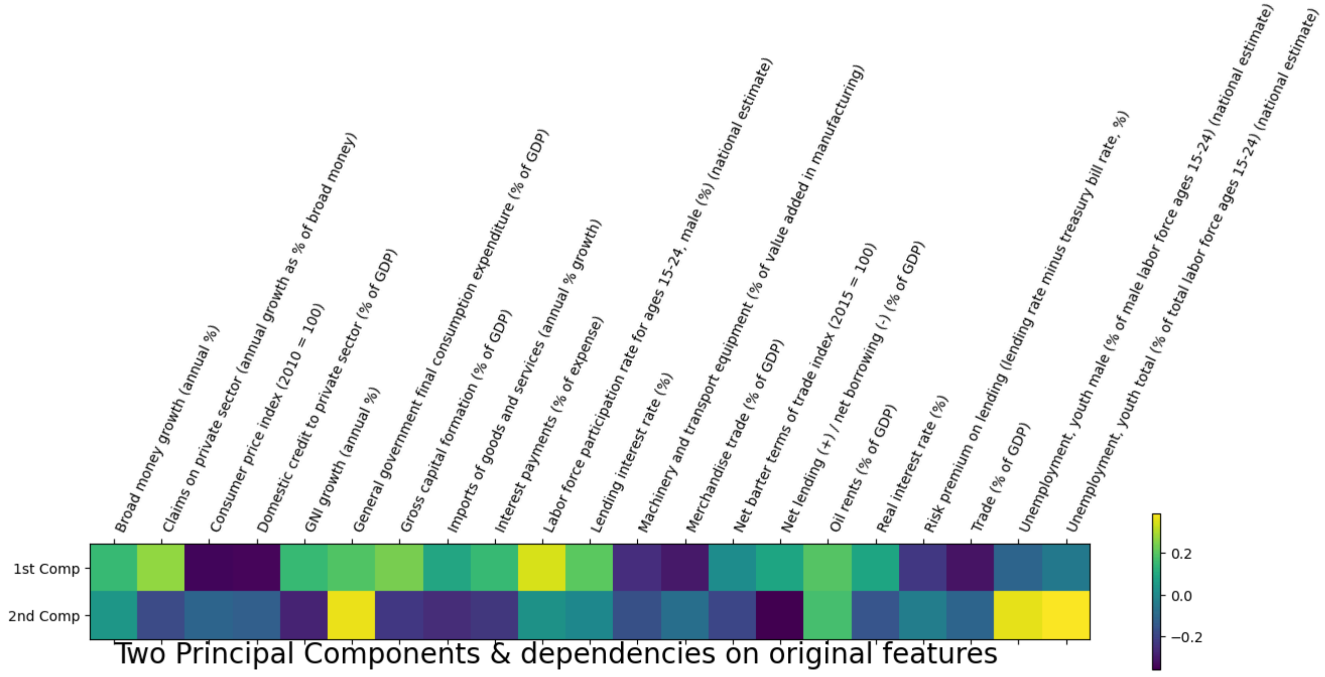

Figure 2: Heatmap of Principal Component Loadings

It can be observed that PC1 captures a significant portion of the variance, suggesting that the labor force participation rate for youth, overall price of commodities, and financial resources provided to both corporations and non-profit organizations indicates a strong relationship with economic downturns. PC2, however, highlights youth unemployment and public spending, suggesting their relevance in the broader economic landscape.

Model Development and Evaluation

Logistic Regression Model Performance:

- Model Trained on Raw Scaled Data:
  - Accuracy: 84.62% (Low: 53%, High: 100%)
- Model Trained on PCA-Transformed Data (3 Components):
  - Accuracy: 92.31% (Low: 61%, High: 100%)

Note: These accuracy values may vary with each run due to the randomness in data splitting.

A difference in accuracy between raw scaled data versus PCA-transformed data on the model can be made, suggesting that reducing dimensions helped the model focus on significant features and enhance predictive performance.

K-Nearest Neighbors (KNN) Model Accuracy: 76.92%. This performance was comparable to logistic regression but highlighted challenges associated with imbalanced datasets.

Random Forest Model Accuracy: 100%. Unfortunately, this performance very strongly suggests the presence of overfitting.

Classifier Visualizations

To visualize the decision boundary of the Logistic Regression classifier, we reduced the data to two principal components and visualized the model's classification of points near the data in the two-component subspace. The scatter plot below (Figure 3) illustrates the model's decision boundary between recession and non-recession classes. The red-shaded region in the plot represents the region of points in the two-component subspace classified as recessions by the model. Conversely, the blue-shaded region in the plot represents the region of points in the two-component subspace classified by the model as non-recessions. The training and test data are displayed and color coded according to the true labels to visualize the accuracy of the model's decision boundary.

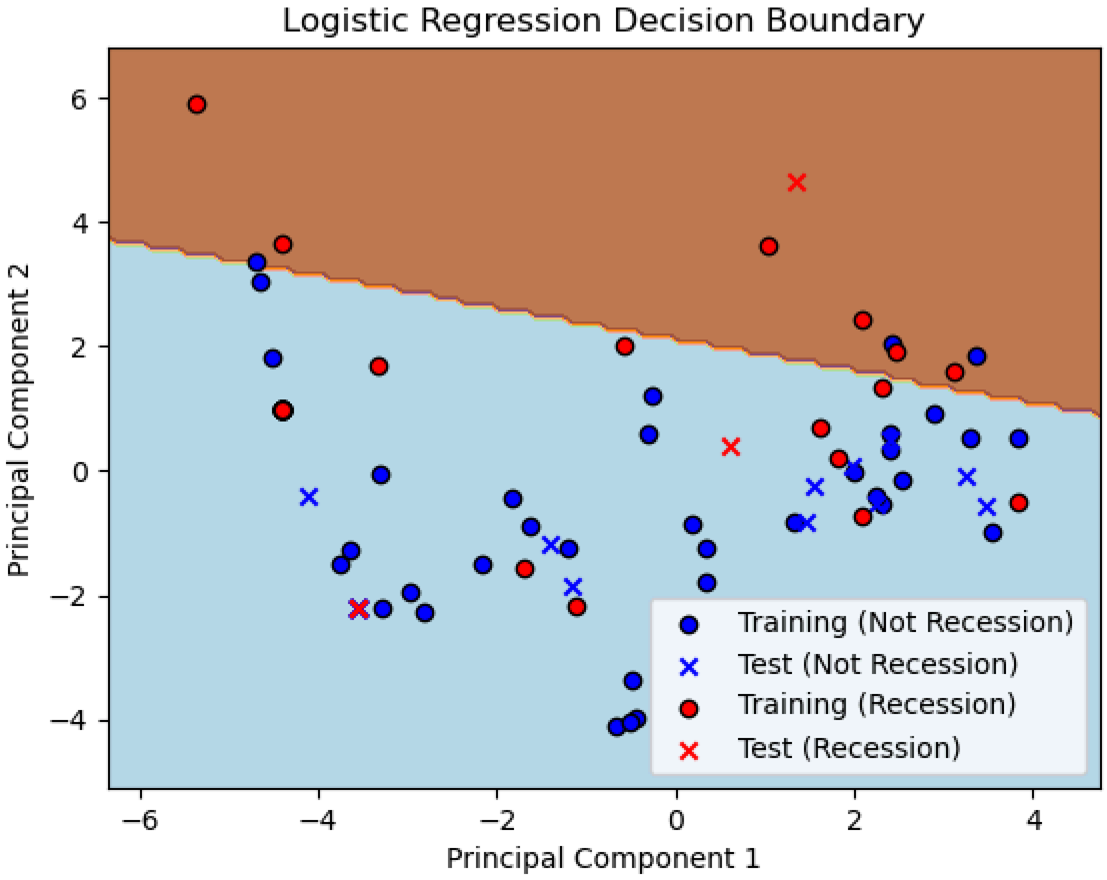

Figure 3: Logistic Regression Decision Boundary

Visualizing the decision region for KNN followed a similar procedure where Figure 4 showcases the PCA-applied scatter plot with k = 18 neighbors. The orange-shaded region in the plot represents the region of points in the two-component subspace classified as recessions by the model. Conversely, the blue-shaded region in the plot represents the region of points in the two-component subspace classified by the model as non-recessions.

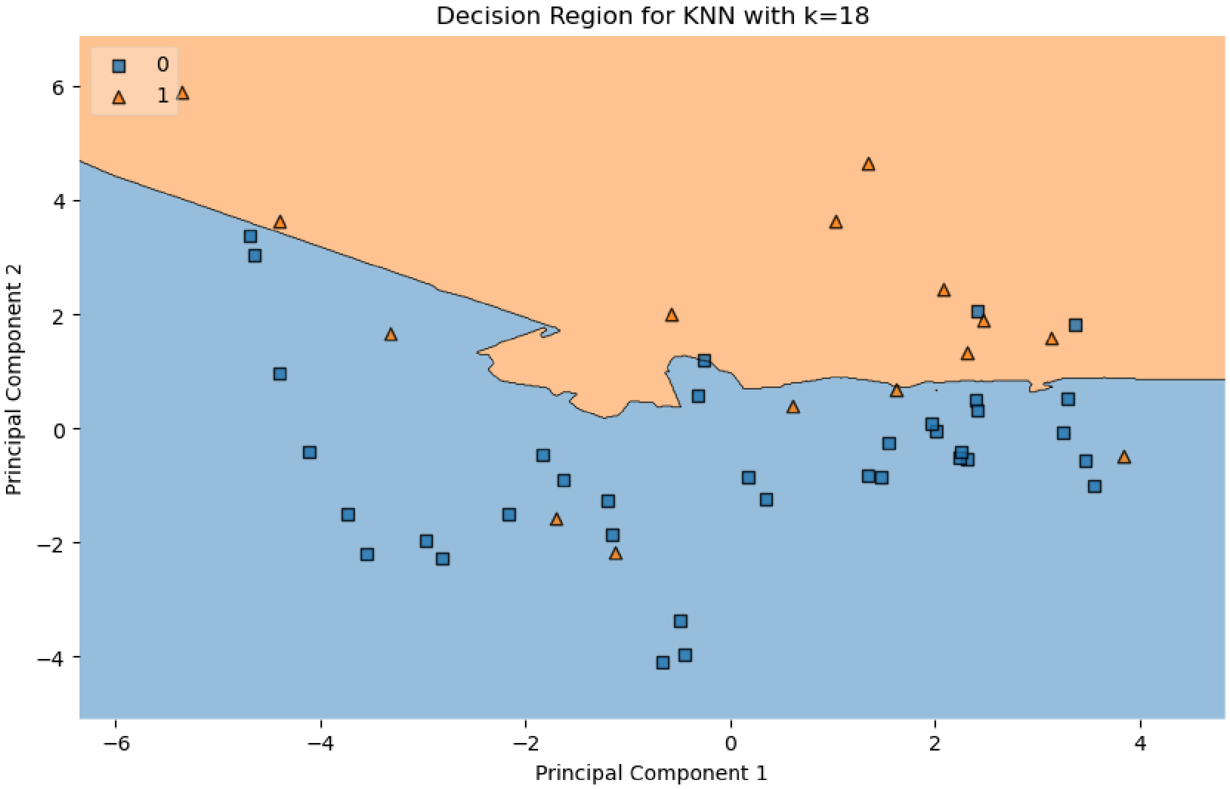

Figure 4: K-Nearest Neighbors Decision Boundary

The Random Forest decision boundary plot in the PCA-transformed space (Figure 5) is illustrated with two principal components. Identical labelling characteristics are present here with an orange decision boundary classifying data points as recessions and blue as non-recessions. Orange triangles and blue squares are true labels for both recessions and non-recessions, respectively. Random Forest also produced a plot of normalized scores containing the top 3 most important features (Figure 6) being *Imports of Goods and Services (annual % growth)* with a score of 0.163, *GNI growth (annual %)* with a score of 0.142, and *Risk premium on lending (lending rate minus treasury bill rate, %)* with a score of 0.101.

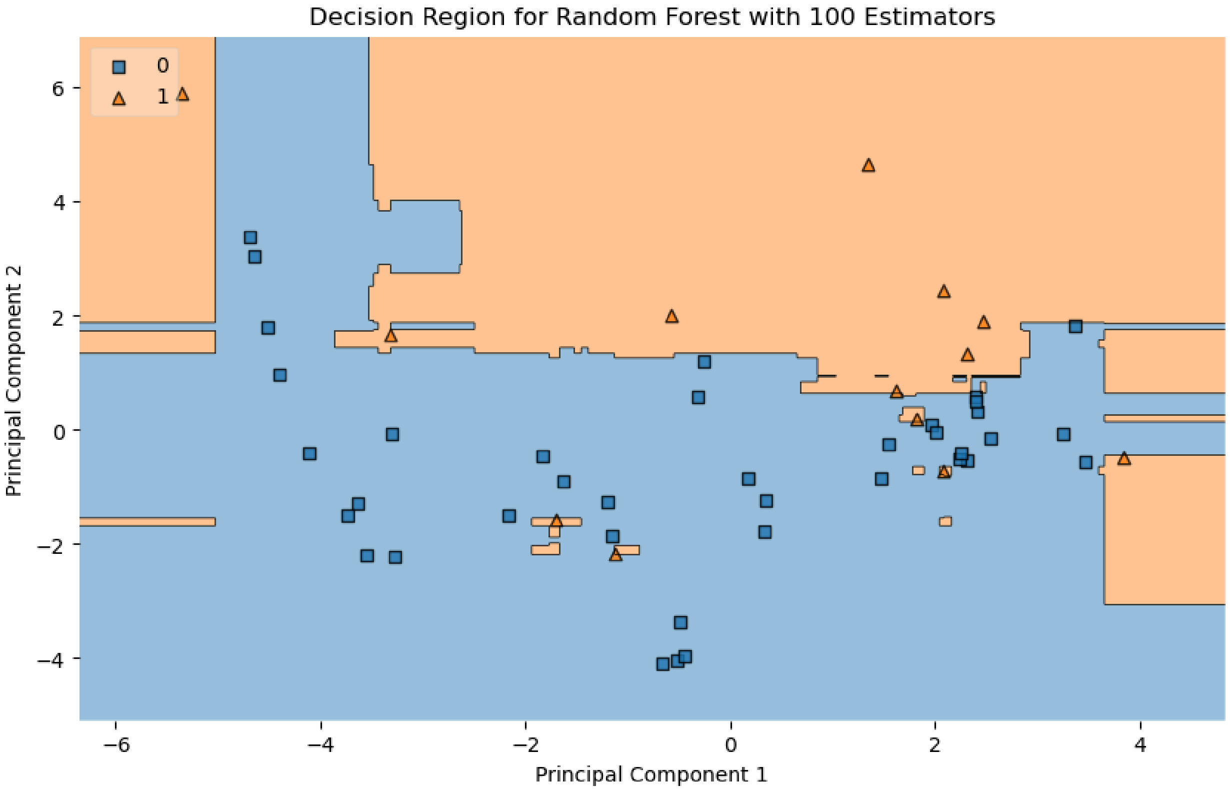

Figure 5: Random Forest Decision Boundary

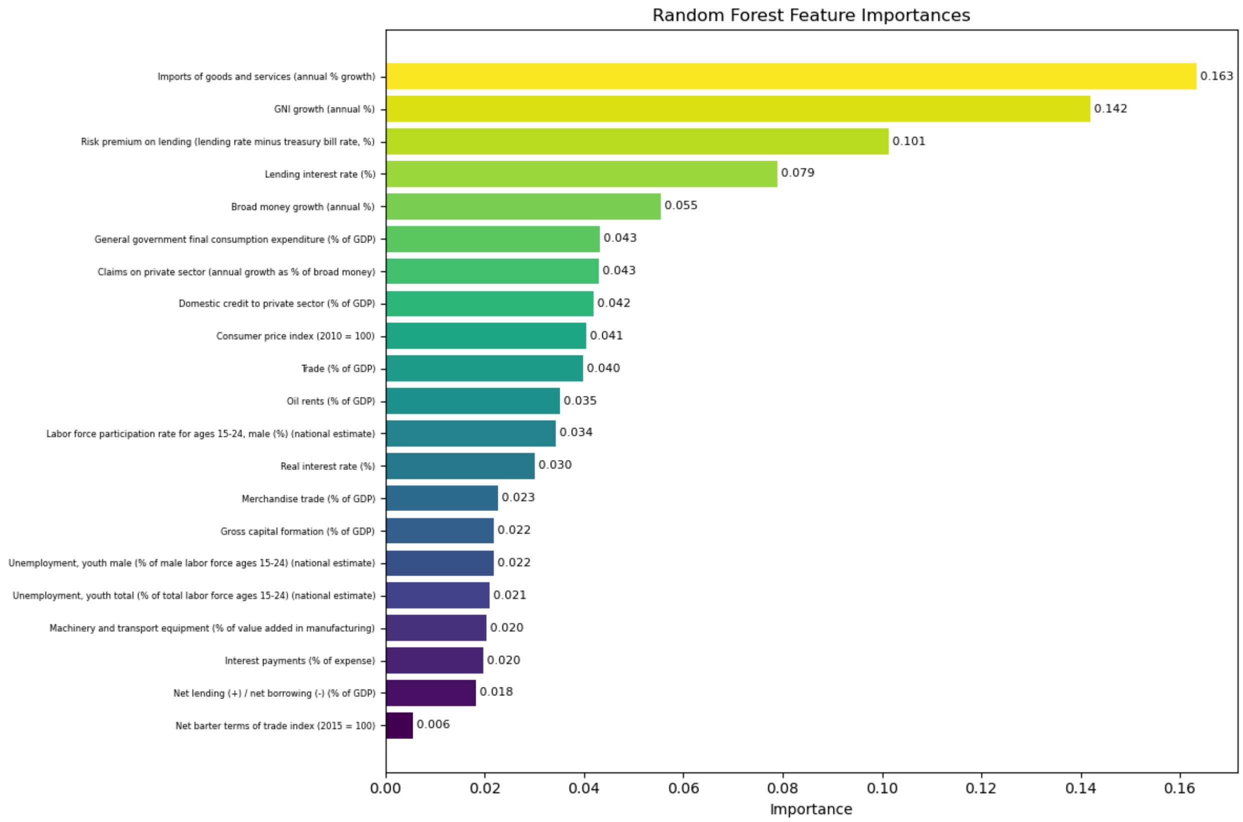

Figure 6: Feature Importances for Random Forest Plotted in Descending Order

Logistic Regression Validation

Cross-Validation:

- Scores: [72.73%, 90.00%, 90.00%, 90.00%, 90.00%]
- Mean Accuracy: 86.55%
- Standard Deviation: 6.91%

An average cross-validation accuracy of 86.55% and standard deviation of 6.91% indicates the model's consistent performance across different folds.

KNN Validation

Cross-Validation:

- Best Cross-Validation Accuracy: 77.81% at k = 20
- Test Set Accuracy: 76.9%

Hyperparameter tuning shown in Figure 7 further validated k = 20 as the most optimal parameter for balancing performance.

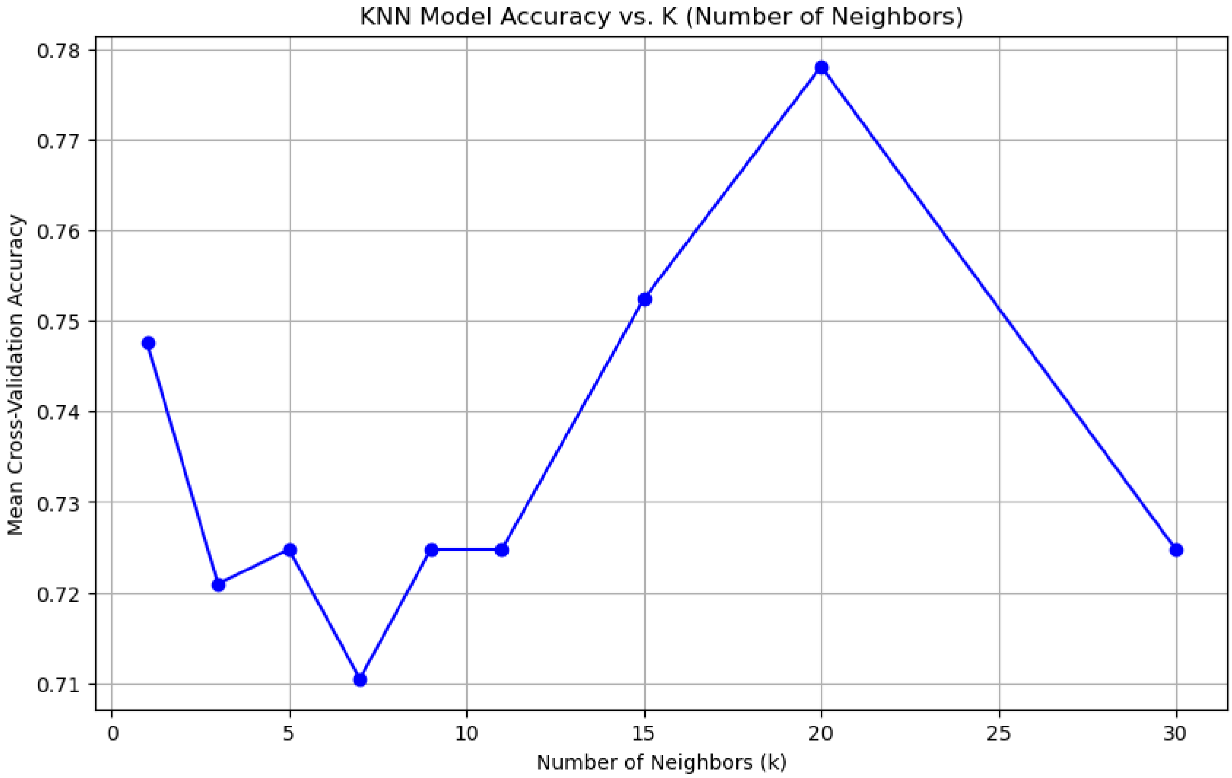

Figure 7: Mean Cross-Validation Accuracy vs. Number of Neighbors (k)

Random Forest Validation

Cross-Validation:

- Best Cross-Validation Accuracy: 88.18% at n = 10 decision trees.
- Test Set Accuracy: 100%

Hyperparameter tuning for the Random Forest model is shown in Figure 8, which further validates the best parameter being n = 10 decision trees.

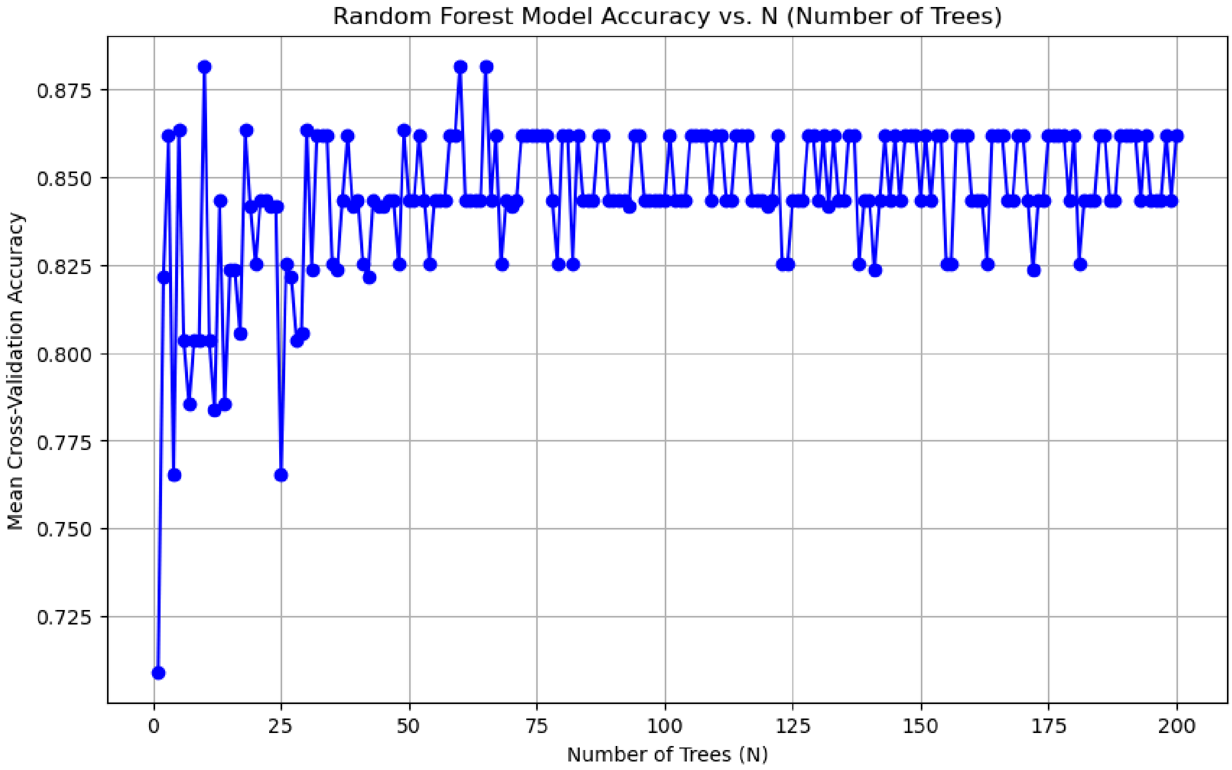

Figure 8: Random Forest Accuracy vs. Number of Trees

Quantitative Metrics (Logistic Regression)

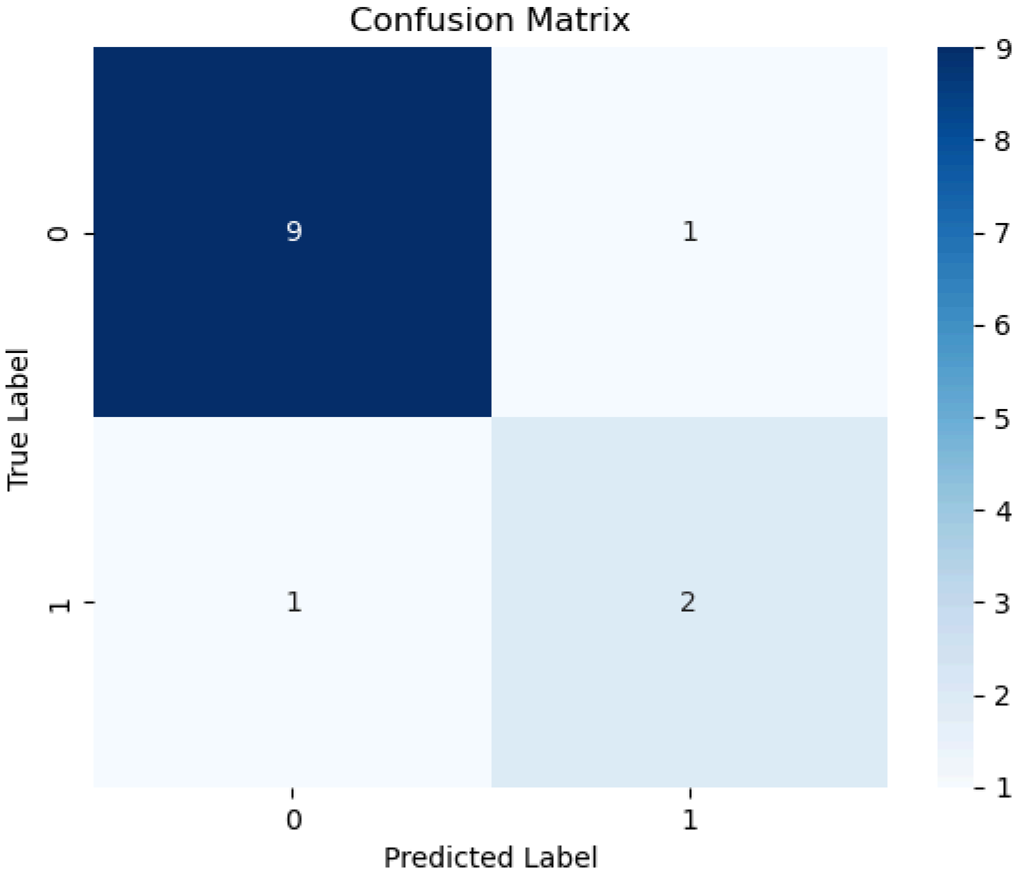

Figure 9: Confusion Matrix (Logistic Regression)

Class	Precision	Recall	F1-Score	Support
0	0.90	0.90	0.90	10
1	0.67	0.67	0.67	3
Accuracy			0.85	13
Macro Average	0.78	0.78	0.78	13
Weighted Average	0.85	0.85	0.85	13

Figure 10: Classification Report (Logistic Regression)

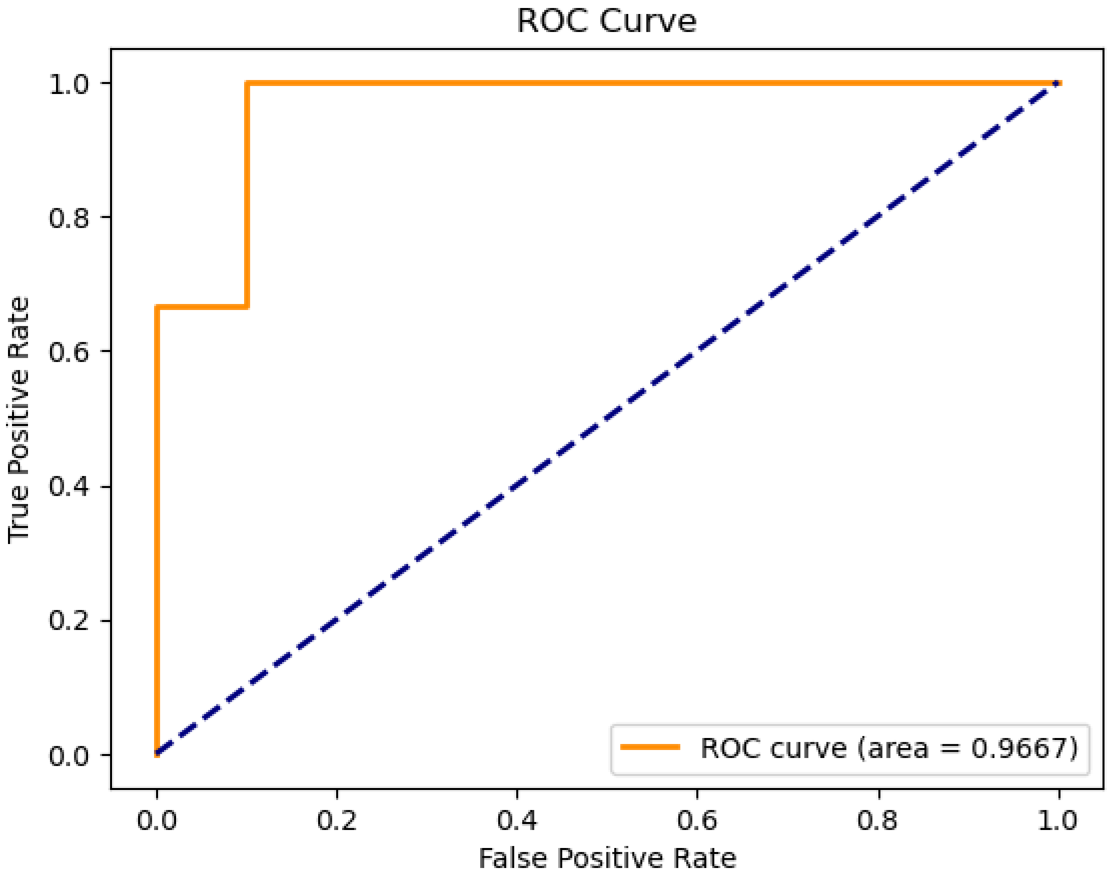

Figure 11: ROC Curve (Logistic Regression)

ROC AUC Score: 0.9667

As indicated in figure 3 and via the classification report, the model correctly classified 90% of non-recession data points but only 67% of recession instances. This could be due to either a class imbalance or there's an inherent complexity in predicting recessions. A high precision, recall, and F1-score for the non-recession class (0) indicate reliability. However, the metrics are much lower for the recession class (1), highlighting challenges in predicting recessions specifically. Given this, the ROC AUC score is suspiciously high at a value of 0.9667. Cross-validation suggests desirable performance, although the model's high accuracy on training, especially given the high ROC AUC score, suggests overfitting which needs to be addressed.

Quantitative Metrics (KNN)

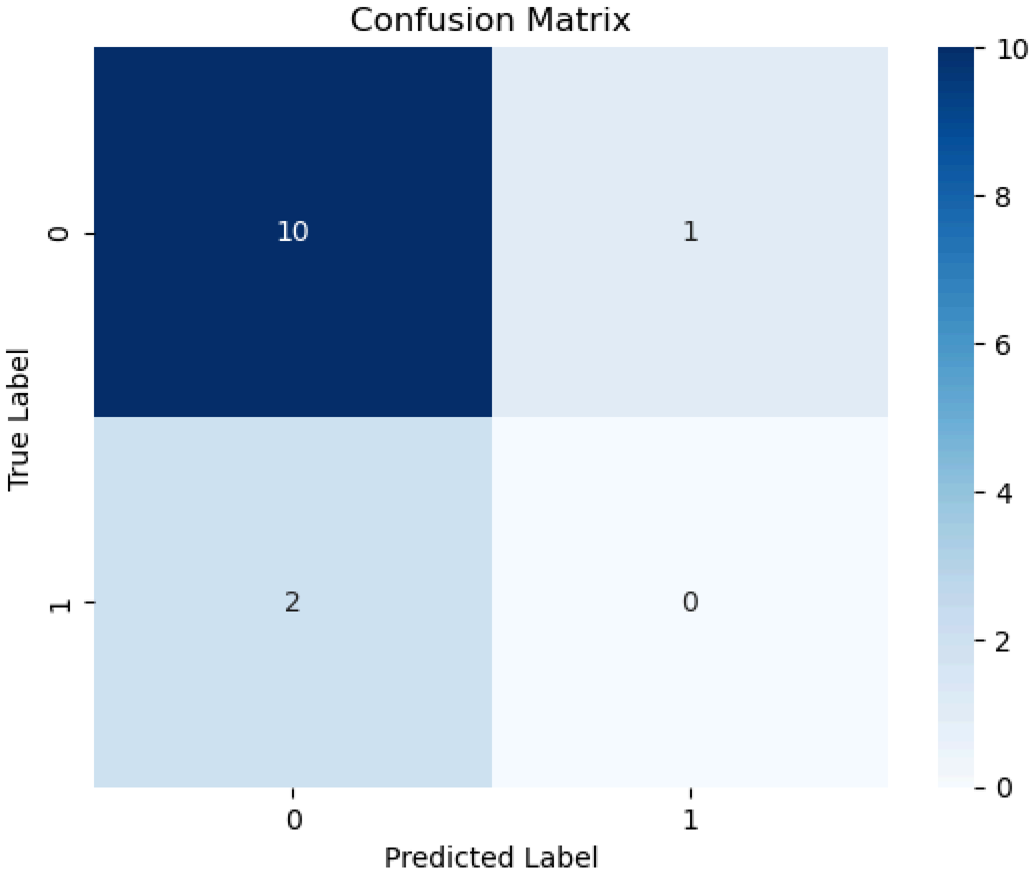

Figure 12: Confusion Matrix (KNN)

Class	Precision	Recall	F1-Score	Support
0	0.83	0.91	0.87	11
1	0.00	0.00	0.00	2
Accuracy			0.77	13
Macro Average	0.42	0.45	0.43	13
Weighted Average	0.71	0.77	0.74	13

Figure 13: Classification Report (KNN)

The confusion matrix reveals that out of 11 non-recession instances (Class 0), only 2 were misclassified as recessions. Specifically, of 11 true instances, the model correctly classified 10 and misclassified 1. However, of the recession instances (Class 1), out of the true instances, the model failed to classify any correctly, with both instances being misclassified as non-recessions. Precision, recall, and F1-scores from the classification report further supports the disparity between class predictions. Non-recessions (0) showed high precision (0.83), recall (0.91), and F1-score (0.87). However, recessions (1) showed no precision or recall, indicating critical underperformance that needs to be addressed to properly classify recessions for this particular model.

Quantitative Metrics (Random Forest)

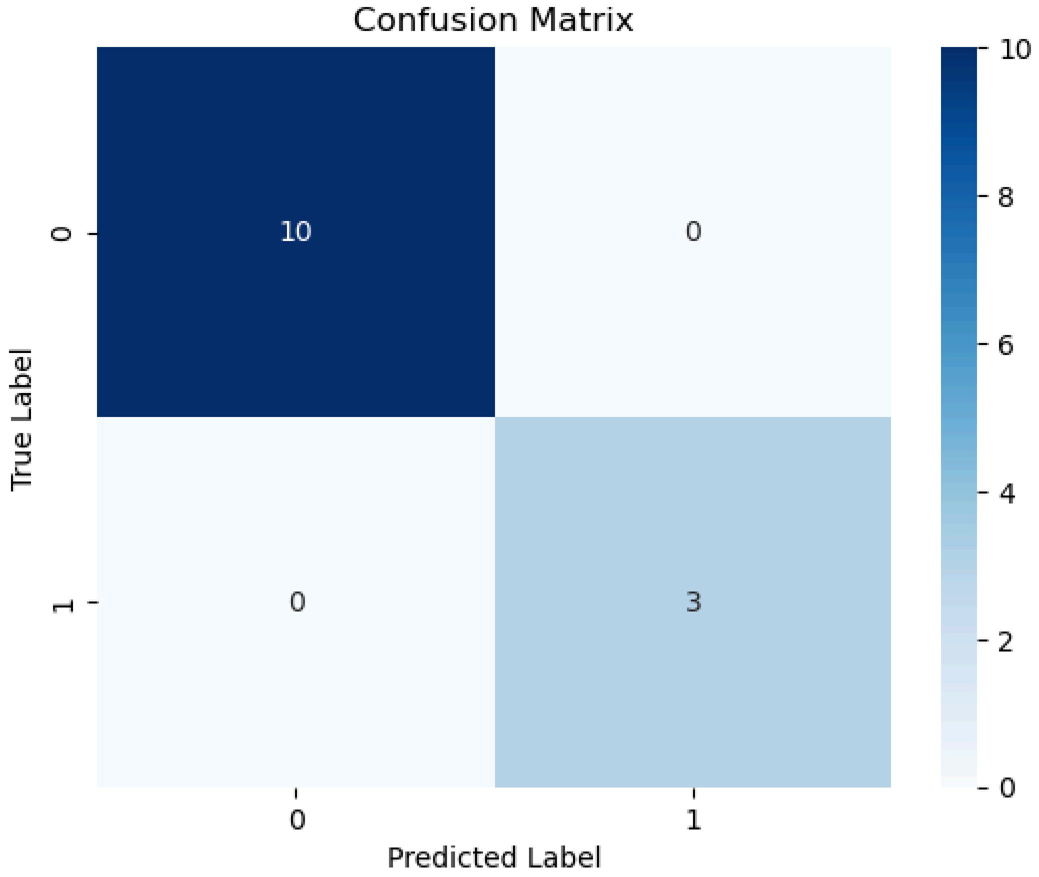

Figure 14: Confusion Matrix (Random Forest)

Class	Precision	Recall	F1-Score	Support
0	1	1	1	10
1	1	1	1	3
Accuracy			1	13
Macro Average	1	1	1	13
Weighted Average	1	1	1	13

Figure 15: Classification Report (Random Forest)

Figures 14 and 15 reveal that for the non-recession class (0), all 10 instances were correctly classified, resulting in perfect precision, recall, and F1-scores. The same holds true for the recession class (1) as well, where all 3 instances were classified accurately. While this performance on the test set is exceptional, showing macro and weighted averages of 1.00 for all metrics, it raises strong concerns about overfitting, especially given the small and imbalanced test set. It is quite possible that the model learned specific patterns in the training data too well, leading to an inability to generalize newer data effectively. Further testing on larger, more balanced datasets is necessary to confirm its true accuracy.

Discussion

The results of the three models–Logistic Regression, K-Nearest Neighbors (KNN), and Random Forest–highlight some successes but significant challenges in applying machine learning techniques to prediction economic recessions. Each model's unique approach to picking out patterns revealed valuable insights while exposing limitations present in our methodology.

Logistic Regression was particularly effective when applied to the PCA-transformed dataset, achieving consistently high accuracy and improved accuracy compared to raw features. Performing dimensionality reduction seems to have improved the model's ability to generalize and focus on the most relevant indicators. However, the model struggled significantly with class imbalance, leading to weaker recall and F1-scores for recession predictions.

Similar issues were encountered when implementing K-Nearest Neighbors, despite recruiting class balancing techniques such as SMOTE and undersampling. Even though the model performed well for non-recessions, it failed to correctly classify any recession instances accurately. This can be attributed to KNN’s sensitivity to imbalanced data distributions. In the future, more refined sampling techniques could be utilized to overcome this disappointing result.

While logistic regression had a somewhat balanced classification rate and KNN purely underperformed recession classification, Random Forest seems to occupy the opposite end of the spectrum with fully accurate classification for both the non-recession and recession class. While this is promising, this very likely indicates overfitting due to the dataset’s small size and imbalance. However, Random Forest’s ability to provide identifiers for the most impactful features, particularly import growths and risk premium on lending, gives valuable insight for future research. While refinement is necessary, this serves as a great starting point for further economic analysis.

A recurring theme present across all our models was the presence of class imbalance in our datasets. Recession periods are rare compared to non-recession periods, leading to an imbalanced dataset with bias skewed towards its majority class without appropriate representation of the minority class in the trained logistic regression model. Economic relations may not be fully captured within our data set and necessary indicators may not be present, reducing prediction accuracy. Techniques such as SMOTE and undersampling were deployed to counteract this discrepancy, however they still fell short of fully addressing the issue.

Next Steps

One approach that could be undertaken to navigate around the class imbalance issue could be simply expanding our dataset to include more recession periods generated via advanced sampling procedures. Additionally, adding features such as housing starts, consumer confidence index, and S&P 500 index would provide a more comprehensive analysis. Alternative modelling approaches could also be explored. Given these outlines, we remain optimistic that machine learning could be used to help policymakers and analysts predict economic downturns.

References

Y. Huang and E. S. Yan, "Economic Recession Forecasts Using Machine Learning Models Based on the Evidence from the COVID-19 Pandemic," *Modern Economy*, vol. 14, no. 7, pp. 899-922, Jul. 2023, doi: 10.4236/me.2023.147049.

K. Tehranian, "Can Machine Learning Catch Economic Recessions Using Economic and Market Sentiments?" University of California – Los Angeles, 2023. [Online]. Available: https://arxiv.org/abs/2308.16200.

N. Zyatkov and O. Krivorotko, "Forecasting Recessions in the US Economy Using Machine Learning Methods," in *17th International Asian School-Seminar on Optimization Problems of Complex Systems (OPCS)*, Novosibirsk, Russia, 2021, pp. 139-145, doi: 10.1109/OPCS53376.2021.9588678

Streamlit:

https://cs4641group84.streamlit.app/

Video:

https://youtu.be/Qz3P5_xCWAo

Gantt Chart

https://gtvault-my.sharepoint.com/:x:/g/personal/lzhou330_gatech_edu/ESEaqEj0k55HvogqSwxle6MBCv2w_OLwNJ0soUL1prCBrg?e=ZfA7Fm

Contribution Table

Name	Proposal Contributions
Sarah Chae	Data Preprocessing (Feature Selection, PCA), KNN implementation
Cordell Alexander Palmer	Visualization, Report Writing, Analysis
Mitchell Brian Teunissen	Logistic Regression implementation (Training/Inference), Decision Boundary Visualization, Random Forest implementation
Jerry Wu	Implementation of Model (Validation/Testing), Video
Lawrence Zhou	Quantitative Metrics, Analysis, Streamlit