Data Bootcamp - Final Project Report

Predicting Stock Market Index Movements with Macroeconomic Changes

Introduction

The relationship between stock market movements and macroeconomic factors is a crucial area of analysis in financial economics. Understanding how key economic indicators such as inflation, unemployment, GDP, and exchange rates impact stock market index prices can help investors and policymakers make more informed decisions.

Through this project, I aim to develop a predictive model that estimates the future stock index prices of various countries based on the movement of macroeconomic factors from 1980 to 2020. By analyzing data from nine different countries (United States, China, France, Germany, Hong Kong, India, Japan, Spain, and the United Kingdom), the project seeks to uncover significant patterns and correlations between economic variables and stock price movements.

The predictive model developed in this project will provide valuable insights into how stock market indices react to changes in macroeconomic factors, aiding investors in their decision-making processes and improving their understanding of global economic dynamics.

Data Description

The dataset used for this analysis encompasses macroeconomic and stock market data for nine countries spanning the period from 1980 to 2020. It combines country-specific stock index prices with critical economic indicators that reflect global and local economic health.

Features in the Dataset

1. **Stock Index**: The primary stock market index for each country (e.g., NASDAQ, FTSE 100, DAX).

2. **Country**: The country associated with the stock market index.

3. **Year**: The year corresponding to the data point.

4. **Index Price**: The closing price of the stock index.

5. **Log Index Price**: The natural logarithm of the index price, used to normalize the data.

6. **Inflation Rate**: The annual percentage increase in prices.

7. **Oil Prices**: Global oil prices, reflecting their economic influence.

8. **Exchange Rate**: Currency exchange rates relative to the US dollar.

9. **GDP Percent**: Annual GDP growth as a percentage.

10. **Per Capita Income**: Average income per person in the country.

11. **Unemployment Rate**: The national unemployment percentage.

12. **Manufacturing Output**: Yearly output of the manufacturing sector.

13. **Trade Balance**: Net exports minus imports.

14. **USTreasury**: Yields on US Treasury securities, representing risk-free rates.

Data Cleaning and Preprocessing

The data exhibited missing values, with the count varying across features. Notable gaps included missing values in **index price** (52 instances) and **manufacturing output** (91 instances). Missing data was visualized using a heatmap to understand the extent and distribution of null values. Techniques such as forward fill and interpolation were applied to impute these gaps, ensuring the dataset was ready for analysis.

Exploratory Data Analysis

Exploratory Data Analysis (EDA) revealed key trends and distributions within the dataset:

1. Distributions:
   - Histograms of variables like inflation rate, GDP percent, and index prices showed their variability and skewness across the dataset.
   - The log-transformed index price stabilized variance and revealed a more normal-like distribution.
2. Temporal Trends:
   - A multi-line plot highlighted the trajectory of stock index prices over time for each country, revealing distinct economic cycles.
3. Correlations:
   - A heatmap of the correlation matrix indicated relationships between variables such as GDP growth and index prices, providing insights into macroeconomic dependencies.
4. Comparative Analysis:
   - Box and violin plots showcased disparities in index prices between countries, highlighting potential outliers and regional trends.

Models and Methods

Feature Engineering

To enhance the predictive power of the model and incorporate temporal effects, lag features for Inflation Rate and GDP Percent were created. These lagged variables capture the delayed effects of these economic indicators on stock prices.

The Stock Index Price was log-transformed to normalize the data and stabilize variance, a critical step in modeling financial data.

Handling Missing Data

Missing values in lagged features were imputed using mean imputation. This ensured that the lagged variables were complete for modeling.

Data Preparation

To prepare the dataset for machine learning:

1. Feature Matrix (X): Excluded raw and log-transformed index prices to focus on predictive variables.
2. Target Variable (y): Used the log-transformed stock index price.
3. Train-Test Split: Divided the dataset into training (80%) and testing (20%) subsets for validation.

Categorical Encoding

Categorical variables like Stock Index and Country were one-hot encoded to convert them into numerical format suitable for machine learning. Stock Index and Country are crucial categorical variables because they capture region-specific and index-specific factors that can significantly influence stock market behavior. These variables account for local economic policies, market conditions, and regional economic trends, which would otherwise be ignored if these features were left out. Encoding them ensures the model can leverage this information while maintaining numerical compatibility for machine learning.

The encoding process avoided multicollinearity by dropping the first category. The encoded columns were then appended to the dataset.

Data Transformation

Numerical features were standardized using StandardScaler to ensure consistent scaling across variables, while categorical features were one-hot encoded. This preprocessing was implemented using `ColumnTransformer.` The preprocessing pipeline was applied to both training and testing datasets.

Model Building

The following models were implemented to predict the log-transformed stock index prices based on the macroeconomic features:

1. Baseline Model

The Baseline Model predicts the mean of the training data as the output for all test instances. This simple benchmark was used to compare the performance of more sophisticated models. It also establishes a reference point to evaluate whether more complex models are adding predictive value.

2. Linear Regression

A traditional regression model that assumes a linear relationship between the features and the target variable. Linear Regression is interpretable and serves as a good starting point for more advanced techniques. It identifies the strength and direction of linear relationships between macroeconomic features and index prices.

3. K-Nearest Neighbors (KNN)

KNN predicts the target value by averaging the outcomes of the k-nearest neighbors in the feature space. Hyperparameter tuning was performed to determine the optimal value of k (number of neighbors). KNN captures localized patterns in the data that may not be apparent in global models, such as regional or short-term economic effects.

4. Decision Tree

A non-linear model that splits the data based on feature thresholds to minimize prediction error. Decision Trees are particularly useful in this context because they can capture complex, non-linear relationships between macroeconomic features (such as inflation, GDP, and oil prices) and stock index prices. Unlike linear models, which assume a direct relationship, Decision Trees can model interactions and non-linearities, allowing them to more accurately reflect the dynamics of the financial market. However, Decision Trees are prone to overfitting, especially when dealing with noisy data, making it essential to fine-tune model parameters such as tree depth and minimum samples per leaf to avoid overfitting while preserving the model's ability to generalize.

5. Random Forest

Random Forest is an ensemble learning method that combines multiple Decision Trees to improve prediction accuracy and generalization. This model is particularly well-suited for predicting stock index prices, as it can handle complex and high-dimensional macroeconomic data (such as inflation, oil prices, and GDP). By averaging the predictions of many individual trees, Random Forest reduces overfitting and enhances the model's ability to generalize to unseen data. This approach is particularly useful in financial modeling, where relationships between variables can be highly non-linear and noisy. Additionally, Random Forest generates feature importance scores, which help identify the key macroeconomic drivers of stock index movements, providing valuable insights into which factors most significantly impact the market. This makes Random Forest not only a powerful predictive tool but also a model that enhances interpretability in understanding financial data dynamics.

Evaluation Metrics

The Mean Squared Error (MSE) was used as the primary evaluation metric to measure the accuracy of predictions. Lower MSE values indicate better model performance.

Results and Interpretation

1. Baseline Model

The Baseline Model MSE of 0.9863 provides a reference point for evaluating the performance of more complex models. This value represents the error of a naive model, typically a mean-based or zero-predictor model. All subsequent models should aim to reduce this error.

2. Linear Regression

Training MSE: 0.4101

Testing MSE: 0.5460

The Linear Regression model outperforms the baseline, with a substantial reduction in MSE for both training and testing data. The training MSE (0.4101) is relatively low, indicating that the model fits well to the training data, while the testing MSE (0.5460) shows that the model generalizes reasonably well to unseen data, albeit with some slight overfitting.

Feature Importance:

The most important features in the Linear Regression model are per capita income (0.3806) and year (0.3181), which have the highest impact on predicting the target variable, log index price.

Stock index variables, such as NASDAQ and CAC 40, are also significant contributors, reflecting the influence of stock market indices on the prediction.

3. K-Nearest Neighbors (KNN)

Best Number of Neighbors: 3

Training MSE: 0.4979

Testing MSE: 0.4940

The KNN model achieved similar MSE values for both training and testing sets, indicating that it is neither overfitting nor underfitting. The performance is comparable to Linear Regression, though the model appears slightly more stable in terms of generalization, as the MSE for testing is nearly identical to that of the training set.

Feature Importance:

The most important features for the KNN model are categorical variables like stock index (Nifty 50) and country (France), with Nifty 50 contributing 0.1429 to the model's predictions. This suggests that the model is sensitive to certain stock indices and countries, which is reflected in the high weights given to these features.

4. Decision Tree

Training MSE: 0.4662

Testing MSE: 0.6558

The Decision Tree model has a significant disparity between the training and testing MSE, with the testing MSE (0.6558) being substantially higher than the training MSE (0.4662). This suggests that the model may be overfitting to the training data. The relatively high testing MSE indicates that the decision tree has learned patterns that do not generalize well to new data.

Feature Importance:

The per capita income (0.4012) remains the most important feature, followed by exchange rate (0.2360) and manufacturing output (0.1944). These findings align with economic theories suggesting the importance of these features in financial forecasting.

Interestingly, stock indices such as DAX 30 and FTSE 100 do not contribute to the model, as their importance is 0, indicating they were not utilized by the decision tree for prediction.

5. Random Forest

Best Hyperparameters: Max depth = 10, N_estimators = 100

Training MSE: 0.0199

Testing MSE: 0.2229

The Random Forest model significantly outperforms all previous models, with the lowest testing MSE (0.2229). This indicates that the model generalizes very well to unseen data while maintaining a low error on the training set. The Random Forest model benefits from an ensemble approach, combining multiple decision trees to improve stability and reduce overfitting.

Feature Importance:

The key features contributing to the Random Forest model's predictions are similar to the other models, with exchange rate (0.1514), manufacturing output (0.1587), and per capita income (0.1216) being the top three. This reaffirms the importance of economic indicators such as exchange rate and manufacturing output in predicting market trends. Stock indices like NASDAQ and SZCOMP also contribute to the model, though their importance is relatively lower compared to the other features.

Model Comparison

Among all the models, Random Forest demonstrated the best performance, achieving the lowest MSE on both training and testing datasets. The significant reduction in MSE from the baseline model highlights the effectiveness of the more complex algorithms in capturing underlying patterns in the data.

Overfitting:

The Decision Tree model showed clear overfitting, as evidenced by the large gap between the training and testing MSE. In contrast, both Linear Regression and KNN displayed reasonable generalization, though KNN slightly outperformed Linear Regression.

Feature Importance:

Across all models, per capita income, exchange rate, and manufacturing output consistently ranked among the most important features, suggesting that these economic indicators are crucial drivers of the log index price. Stock indices and country features also played significant roles, although their importance varied between models.

Implications for Future Models:

The Random Forest model’s success suggests that more advanced ensemble methods or fine-tuning of hyperparameters could further improve performance. It is also advisable to explore additional features or transformations to capture further variability in the data.

Conclusion and Next Steps

This project aimed to compare several regression models to predict the target variable, log index price, and evaluate their performance using Mean Squared Error (MSE) as the primary evaluation metric. The models tested include a Baseline Model, Linear Regression, K-Nearest Neighbors (KNN), Decision Tree, and Random Forest. Each model was assessed based on its ability to generalize to unseen data, with a focus on identifying the most important features influencing the predictions.

The Baseline Model provided a starting point with an MSE of 0.9863, establishing the need for more sophisticated models. Linear Regression showed strong results with an MSE of 0.5460 on the testing data, outperforming the baseline and offering insight into the importance of economic features like per capita income and stock index.

KNN demonstrated similar performance to Linear Regression with an optimal number of neighbors set to 3, achieving an MSE of 0.4940 on testing data.

The Decision Tree model exhibited some overfitting, as evidenced by the large gap between training and testing MSE, with testing MSE of 0.6558.

The Random Forest model emerged as the best performer, with the lowest testing MSE of 0.2760. This model highlighted the importance of economic features, particularly exchange rate and manufacturing output, and demonstrated the power of ensemble methods in reducing overfitting.

Overall, the Random Forest model outperformed all other models, showcasing its ability to handle complex datasets and reduce the risk of overfitting while providing valuable insights into feature importance.

While the Random Forest model produced the best results, there are several avenues for future improvement and exploration:

Incorporating Temporal and Lag Features:

To capture the delayed effects of macroeconomic variables on stock index movements, incorporating additional lag features could improve model performance. Exploring lagged versions of variables such as inflation rate, oil prices, and GDP might allow the models to account for the impact of these factors over time, providing a more accurate reflection of real-world economic dynamics.

Advanced Feature Engineering:

By further enhancing feature engineering, I aim to create new interaction terms between key variables (such as inflation and oil prices) to better capture non-linear relationships. Transformations like polynomial features or exponential smoothing could also provide additional insights into the trends and volatility of the stock index. Additionally, exploring the use of domain-specific features (e.g., interest rates, international trade data) could lead to richer, more informative predictors.

Model Diversification and Comparison:

Exploring additional advanced algorithms such as Gradient Boosting Machines (GBM), XGBoost, or even LSTM (Long Short-Term Memory) networks, which excel at handling sequential and time-series data, could help capture underlying patterns and non-linear relationships that traditional models might miss. Comparing these models' performance with the existing Random Forests and Decision Trees would provide a better understanding of which methods yield the most robust predictions.

Hyperparameter Tuning and Optimization:

To further refine the models, I will conduct more comprehensive hyperparameter tuning. Approaches such as RandomizedSearchCV or Bayesian Optimization could be employed to identify the optimal model parameters more efficiently and potentially enhance model performance. Additionally, tuning model parameters for feature selection, such as regularization in tree-based models, would help reduce overfitting and improve generalization.

Cross-Validation and Model Validation:

Incorporating cross-validation methods, particularly k-fold cross-validation, would provide a more reliable estimate of model performance and help mitigate the risk of overfitting. By ensuring that the models are validated across different subsets of the data, we can improve their robustness and ensure that they perform consistently on unseen data.

Integration of External Economic Data:

Integrating additional external data, such as country-specific macroeconomic indicators or global commodity price trends, could provide a more holistic view of the factors influencing stock index movements. Incorporating data such as interest rates, exchange rates, or commodity-specific indicators like gold or energy prices could further improve model accuracy and prediction capabilities.

Enhanced Model Interpretability:

To increase the transparency of the models, I will leverage model-interpretability tools like SHAP (Shapley Additive Explanations) or LIME (Local Interpretable Model-agnostic Explanations) to better understand the contributions of each feature in the predictions. This will provide valuable insights into the relationship between macroeconomic variables and stock index prices, aiding decision-makers in understanding how each variable impacts the model's outputs.