Gold Price Prediction

A PROJECT REPORT

SUBMITTED TO

SVKM'S NMIMS (DEEMED- TO- BE UNIVERSITY)

IN PARTIAL FULFILLMENT FOR THE DEGREE OF

BACHELORS OF SCIENCE

IN

APPLIED MATHEMATICAL COMPUTING

BY

DIYANSHI SHAH and DIYA GUPTA

Under the guidance of

Dr. Debasmita Mukherjee

NILKAMAL SCHOOL OF MATHEMATICS, APPLIED STATISTICS & ANALYTICS

NMIMS NSOMASA

Ground Floor, SBMP Phase I,

Irla, N. R. G Marg, Opposite Cooper Hospital,

Vile Parle (West), Mumbai – 400 056

November 2024

0.1 Certificate

This is to certify that work described in this thesis entitled “Gold Price Prediction" has been carried out by Diyanshi Shah and Diya Gupta under my supervision. I certify that this is his/her bonafide work. The work described is original and has not been submitted for any degree to this or any other University.

Date:
Place:

SUPERVISOR
Dr. Debasmita Mukherjee

0.2 Acknowledgement

We would like to express our heartfelt gratitude to Dr. Debasmita Mukherjee for her invaluable guidance and mentorship in our Gold Price Prediction project. Her guidance not only deepened our comprehension of the topic but also motivated us to explore the complexities of forecasting gold prices through machine learning models.
Furthermore, we want to express our gratitude to Dr. Subhash Kumar, whose work over various platforms helped in adding depth and quality to the research. This project would not have been possible without the collaborative efforts of all those mentioned above, and we look forward to continuing our work to deliver the best results possible.

Diyanshi Shah and Diya Gupta

(B.Sc. Applied Mathematical Computing, A010 and A002)

0.3 Contents

0.4 Abstract

Accurate forecasting is made extremely difficult by the numerous economic factors that have a considerable impact on gold prices, such as inflation, interest rates, and sentiment in the worldwide market. In order to forecast future gold prices and assess their relative performance and accuracy, this study investigates the use of two machine learning models: Random Forest (RF) and Long Short-Term Memory (LSTM). Key economic variables that have a major impact on changes in the price of gold are identified by analyzing feature importance using the Random Forest model, a tree-based ensemble technique. On the other hand, long-term temporal dependencies are captured using the LSTM model, a kind of recurrent neural network (RNN) that is well-known for its ability to handle time-series data.

The results of the experiments show the unique advantages and disadvantages of each model, emphasizing which strategy provides the most accuracy and dependability for predicting future gold prices. For investors and scholars looking for reliable techniques for predicting the price of gold, this comparative analysis offers insightful information.

0.5 Introduction

Gold, an internationally acclaimed asset, holds a special place as a commodity and an investment vehicle. It is frequently employed as a hedge against inflation and economic uncertainty. Numerous variables, including macroeconomic indices like GDP, interest rates, inflation rates, and investor sentiment, influence changes in gold prices. Predicting gold prices accurately is crucial for well-informed investment and policymaking, but it is still difficult because of the intricate relationships between affecting factors and the inherent volatility.

Time-series forecasting is undergoing inventive techniques made possible by recent developments in machine learning, especially for financial assets like gold. The Random Forest model is one of these methods that has become popular because of its capacity to manage high-dimensional data and carry out feature selection, which helps to discover the economic variables that have the greatest influence on gold price predictions. As for tasks involving temporal dependencies, Long Short-Term Memory (LSTM) networks, a subset of recurrent neural networks, are excellent at identifying nonlinear patterns and sequential relationships in time-series data.

In this research, we evaluate the accuracy with which the Random Forest and LSTM models predict future gold prices. The Random Forest model is employed to evaluate the significance of features and offer insights into the economic factors that have most influence on movements in the price of gold. In contrast, the LSTM model uses the historical data's temporal relationships to provide predictions based on trends across time. This study attempts to ascertain which approach is more appropriate for gold price prediction by assessing the efficacy and accuracy of each model, providing a useful comparison for analysts and investors looking for trustworthy forecasting tools.

0.6 Rationale

Gold is a globally significant commodity, functioning both as an investment vehicle and as a hedge against economic uncertainty. Understanding and forecasting the price of gold is of interest to investors, policymakers, and economic analysts. Gold prices are influenced by multiple macroeconomic factors, including Treasury yields, inflation rates, and GDP growth. These factors reflect the broader economic environment, such as inflationary pressures, interest rate trends, and market volatility, all of which play a key role in gold's value fluctuations.

Given the complexity and interconnected nature of these variables, machine learning models—especially deep learning models like Long Short-Term Memory (LSTM) networks and traditional approaches such as Random Forest—offer an advanced way to capture nonlinear relationships and time-dependent patterns in the data. The aim of this study is to leverage these machine learning techniques to predict future gold prices by incorporating historical gold data and macroeconomic indicators like Treasury yields, inflation, and GDP.

0.7 Aims & Objectives

Aim: Developing machine learning models to forecast the price of gold in the future using historical gold prices and a variety of macroeconomic indices (such as GDP, inflation, and Treasury rates) is the main goal of this research. The models will assist in ascertaining the degree to which these elements influence transient fluctuations in the price of gold.

0.7.1 Objectives:

- Develop an Accurate Gold Price Prediction Model: Leverage machine learning algorithms, such as Random Forest and Long Short-Term Memory (LSTM) models, to accurately predict future gold prices based on historical data and relevant economic indicators.

- Incorporate Economic Indicators: Integrate key economic factors, including treasury yields, inflation, and GDP data, into the predictive models to enhance the forecasting accuracy and capture the broader macroeconomic influences on gold prices.

- Compare Machine Learning Techniques: Evaluate and compare the performance of traditional machine learning methods (Random Forest) against deep learning models (LSTM) to identify the most effective approach for predicting gold price trends over time.

- Detect and Mitigate Overfitting: Implement regularization techniques like Ridge and Lasso to address potential overfitting in predictive models, ensuring robust performance across different datasets.

- Create a User-Friendly Prediction Tool: Develop a streamlined web-based interface that enables users to access gold price predictions, visualize model outputs, and understand the underlying economic trends influencing future prices.

0.8 Literature Review

Gold price prediction has been a subject of substantial interest in financial and academic research. Historical studies have identified that gold serves as a safe-haven asset during periods of financial instability, and its price is largely influenced by factors such as interest rates, inflation, exchange rates, and broader economic growth indicators.

0.8.1 Previous Studies:

- Studies such as Baur & Lucey (2010) have highlighted gold's role as a safe haven, particularly during periods of financial stress. They emphasize the importance of incorporating economic indicators like interest rates into gold price forecasting models.

- Zhang et al. (2017) used deep learning models, including LSTM, to predict financial time series, demonstrating the model's effectiveness in capturing long-term dependencies and non-linear patterns.

- A study by Tang et al. (2021) titled "Prediction of financial time series using LSTM and data denoising methods" explores the application of Long Short-Term Memory (LSTM) networks combined with data denoising techniques for financial time series forecasting.

0.9 Research Methodology

This section outlines the detailed methodology used to achieve the study's objectives, including data collection, preprocessing, exploratory analysis, machine learning model development, and model evaluation.

0.9.1 About the dataset:

- Open: The price at which the gold futures contract started trading when the market opened for that specific day. This value reflects the first trade of the day and is an important indicator of market sentiment.

- High: The highest price reached by the gold futures contract during the trading day. It indicates the maximum price that traders were willing to pay for the futures contract within the trading session.

- Low: The lowest price reached by the gold futures contract during the trading day. It reflects the minimum price that traders were willing to accept during the session, providing insight into the intraday market volatility.

- Close: The final price at which the gold futures contract traded at the close of the market for the day. It is one of the most important prices in a trading session and is widely used as a benchmark for the asset's performance on that day.

- Adj Close (Adjusted Close): The closing price of the gold futures contract, adjusted for any corporate actions like dividends, splits, or new contracts. For commodities like gold, it adjusts for certain market conditions. It gives a more accurate reflection of the asset's value over time, especially when comparing historical data across different time periods.

0.9.2 Data Features

The gold price data is downloaded using yfinance for the symbol 'GC=F' (Gold Futures). Several technical indicators and features are created from this historical data to help the machine learning model make predictions. These features include:

- Close_Diff: Difference between the adjusted closing price of the current and previous day (Adj Close.diff()). This helps capture daily price changes.

- Moving Averages (MA): MA200, MA100, MA50, MA26, MA20, MA12: These are simple moving averages of the closing prices over different time windows (e.g., 200-day, 100-day, etc.). Moving averages help smooth out price data and identify trends.

- SMA Differences: DIFF-MA200-MA50, DIFF-MA200-MA100, etc.: These features represent the difference between different moving averages (e.g., 200-day vs. 50-day moving average).

  These helps capture momentum and crossovers, which are often used as trading signals.

- Highs and Lows: MA200_low, MA14_low, MA200_high, MA14_high: These capture the minimum and maximum values over different periods, helping detect support and resistance levels.
- Standard Deviation: MA20dSTD: The standard deviation of closing prices over 20 days, used in volatility calculations.
- Exponential Moving Averages (EMA): EMA12, EMA20, EMA26, EMA100, EMA200: These are exponentially weighted moving averages, which give more importance to recent data points.
- Shifted Prices: close_shift-1, close_shift-2: These are closing prices from one and two days before, helping the model learn patterns from recent days.

0.9.3 External Data Sources

The code also incorporates additional economic factors that could influence gold prices, including Treasury yields, GDP, and inflation rates.

- Treasury Yields: The yields for different U.S. Treasury bonds are fetched using the following symbols:

    * ˆTNX: 10-Year Treasury Yield
    * ˆIRX: 13-Week Treasury Yield
    * ˆFVX: 5-Year Treasury Yield
    * ˆTYX: 30-Year Treasury Yield

– Treasury yields: The yields for different U.S. Treasury bonds are fetched using the following symbols:

ˆTNX : 10-Year Treasury Yield

ˆIRX : 13-Week Treasury Yield

ˆFVX : 5-Year Treasury Yield

ˆTYX : 30-Year Treasury Yield

These yields are important as they influence investor sentiment, especially for safehaven assets like gold. When bond yields rise, gold might become less attractive because bonds provide better returns.

- GDP: Gross Domestic Product (GDP) data is loaded from an external CSV file and merged with the gold data based on the date. GDP can serve as a proxy for the overall economic health of a country, influencing gold prices indirectly.
- Inflation: Inflation data is loaded from another external CSV file. Inflation affects the purchasing power of money, and higher inflation often leads to higher gold prices as investors use gold to hedge against inflation.

0.9.3 Feature Merging and Preprocessing

- Data Merging: The gold price data is merged with the treasury yields, GDP, and inflation data on the Date column. The resulting dataset combines technical indicators from gold price data with macroeconomic factors.
- Missing Data Handling: The merged dataset is filled forward (ffill()) and backward (bfill()) to ensure there are no missing values, followed by conversion of columns to numeric values.

0.9.4 Exploratory Data Analysis (EDA):

Exploratory Data Analysis (EDA) was conducted to gain an understanding of the dataset, identify patterns, detect anomalies, and uncover relationships between the features. The following steps detail the EDA performed on the gold price and macroeconomic indicator data:

- Summary Statistics:
  - We generated descriptive statistics using the gold.describe() function. This provided key insights into the distribution of the gold prices (such as mean, median, and variance) and other financial indicators (Open, High, Low, etc.).

  Example insights:

    * Mean and standard deviation of the gold price over the 10-year period.

* Minimum and maximum prices indicating the price range within the dataset.

	Open	High	Low	Close	Adj Close	Volume
count	2513.000000	2513.000000	2513.000000	2513.000000	2513.000000	2513.000000
mean	1567.242379	1574.738043	1559.968405	1567.391324	1567.391324	5252.565062
std	359.631605	361.835861	357.759053	359.915213	359.915213	29190.590192
min	1053.699951	1062.000000	1046.199951	1050.800049	1050.800049	0.000000
25%	1251.900024	1256.099976	1247.500000	1251.699951	1251.699951	46.000000
50%	1485.900024	1491.000000	1479.900024	1484.000000	1484.000000	173.000000
75%	1846.099976	1855.800049	1838.699951	1847.800049	1847.800049	536.000000
max	2679.899902	2683.399902	2673.699951	2676.600098	2676.600098	386334.000000

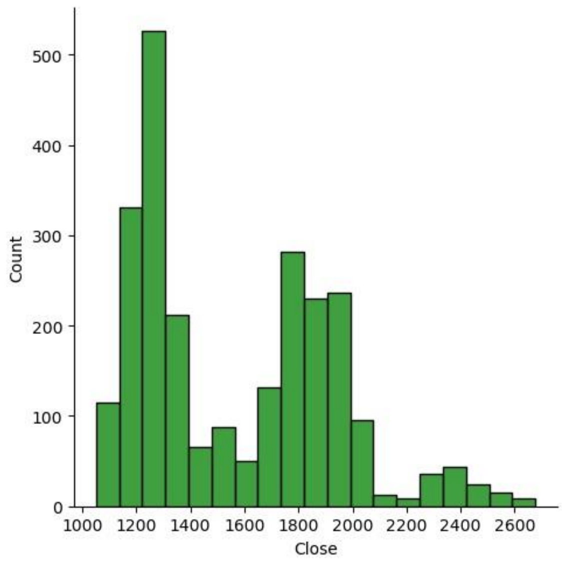
0.9.5 Data Overview:

- The gold.info() function was used to inspect the structure and data types within the dataset. This allowed us to verify which columns were numeric and check for any missing data.
- This step helped confirm that most of the data was numerical, except for the Date column, which needed conversion to a datetime object.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2513 entries, 0 to 2512
Data columns (total 7 columns):
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   Date       2513 non-null   datetime64[ns]
 1   Open       2513 non-null   float64
 2   High       2513 non-null   float64
 3   Low        2513 non-null   float64
 4   Close      2513 non-null   float64
 5   Adj Close  2513 non-null   float64
 6   Volume     2513 non-null   int64
dtypes: datetime64[ns](1), float64(5), int64(1)
```

0.9.5.1 Distribution of Gold Prices:

- The distribution of gold prices (Close) was visualized using a histogram. This plot helped assess the spread and skewness of the gold price data.
- Outcome: The distribution was important for understanding whether the data was normally distributed or skewed, which could influence model selection and performance.



0.9.5.2 Handling Missing Data:

- Although part of data preprocessing, in EDA we initially checked for missing values using gold.isnull().sum(). Identifying missing data early on allowed us to decide how best to handle it, particularly using methods such as forward-fill (ffill) and backward-fill (bfill).

- Outcome: Missing values were dealt with during preprocessing.

```
Date         0
Open         0
High         0
Low          0
Close        0
Adj Close    0
Volume       0
dtype: int64
```

0.9.6 Data Preprocessing:

The preprocessing steps ensured the data was in the right format and cleaned for further analysis and model training.

- Data Splitting:

  The dataset was split into training and testing sets (80% training, 20% testing) using train_test_split(). This step was necessary to train the model on one portion of the data while evaluating its performance on unseen data.

- Overfitting and Regularization in Data Preprocessing

  Overfitting happens when a machine learning model captures the noise or random fluctuations in the training data, resulting in a model that performs well on the training set but poorly on new, unseen data. This is a major issue because the model fails to generalize beyond the data it was trained on.

  To address overfitting, regularization techniques are often applied during data preprocessing. These techniques introduce a penalty for model complexity, encouraging simpler models that generalize better to new data. Two common regularization methods are Ridge Regression and Lasso Regression:

  - Ridge Regression (L2 Regularization): Ridge regression adds a penalty term proportional to the square of the coefficients. This discourages the model from having excessively large coefficients, which can lead to overfitting. By controlling the model's complexity, Ridge regression allows it to generalize better, preventing it from becoming too dependent on the training data.
  - Lasso Regression (L1 Regularization): Lasso regression penalizes the absolute values of the coefficients, which has the added effect of shrinking some coefficients to zero. This means Lasso can automatically select important features, reducing the feature set and simplifying the model. Lasso helps in preventing overfitting by focusing the model on only the most relevant predictors, effectively reducing model complexity and improving generalization.

- Modelling Process

– Features (X) and Target (y): The features for the model (X) exclude the Date and Close columns, as the Close column is the target (the gold price we want to predict). The other columns represent the technical indicators and macroeconomic features.

- Model: A Random Forest Regressor is used to model the relationship between the features and the gold price. Random Forests are an ensemble learning method, combining multiple decision trees to improve accuracy.
- Model Training: The dataset is split into training and test sets using an 80-20 split, and the model is trained on the training data.
- Model Evaluation: The model's performance is evaluated using the R^2 score, which measures how well the model explains the variance in the gold prices.

0.9.7 Model Development:

- Develop a Long Short-Term Memory (LSTM) model that works well with time-series data to forecast gold prices based on historical data and macroeconomic variables.
- As a baseline machine learning approach, train a Random Forest model to capture the significance of several macroeconomic factors in gold price prediction.

0.9.8 Model Evaluation Validation:

When comparing model performance, each evaluation metric provides distinct insights into the model's prediction accuracy. Here's how they differ:

- Mean Absolute Error (MAE): MAE calculates the average magnitude of the errors in a set of predictions, without considering their direction. It gives equal weight to all errors, making it easy to interpret since it reflects the average prediction error in the same units as the data. Usage: Suitable when you want to know the average size of errors and are indifferent to whether the errors are positive or negative.

- Mean Squared Error (MSE): MSE is similar to MAE but squares the errors before averaging, which penalizes larger errors more heavily than smaller ones. This makes MSE more sensitive to outliers. Usage: Often used when larger errors are particularly undesirable, but its interpretation can be less intuitive due to the squaring of units.

- Root Mean Squared Error (RMSE): RMSE is the square root of MSE, bringing the error measure back to the same units as the target variable. Like MSE, RMSE penalizes large errors more heavily, but it is more interpretable since it's in the original scale of the data. Usage: Commonly used in fields where large errors are particularly problematic, and you want to evaluate the magnitude of these errors directly.

- R^2 Score (Coefficient of Determination): R^2 represents the proportion of the variance in the dependent variable that is predictable from the independent variables. An R^2 of 1 indicates perfect prediction, while an R^2 of 0 means the model does not explain any of the variance. Usage: Ideal for determining how well the model explains the variability of the response data around its mean.

- Mean Absolute Percentage Error (MAPE): MAPE measures the size of the error in percentage terms. It is useful for comparing errors across datasets with different scales or for models where the actual values vary significantly. However, MAPE can be misleading when the true values are close to zero, as it can produce very large percentage errors. Usage: Commonly used when you are interested in percentage-based accuracy, but should be used with caution if the data has zero or very small values.

0.10 Machine Learning Models

In this analysis, we applied two machine learning models—Random Forest and Long ShortTerm Memory (LSTM) networks—to predict future gold prices. The study utilized a 10-year dataset of gold prices along with economic indicators such as U.S. Treasury yields, GDP, and inflation rates. Each model was tested to assess which provides greater accuracy in price prediction.

0.10.1 Data Collection and Preprocessing

- Gold Prices: Historical gold prices were collected through Yahoo Finance's API (yfinance) for a 10-year period with daily closing prices. This data provided a foundational time series to model.
- Economic Indicators: Additional financial data, including Treasury yields (10year, 13-week, 5-year, and 30-year), GDP, and inflation rates, were integrated to enhance prediction accuracy. Economic indicators were downloaded from Yahoo Finance and merged on a daily timescale with the gold price data.
- Target and Features: After aligning on the date, the 'Close' column was dropped, and the resulting features included all available indicators (Treasury yields, GDP, and inflation) and technical information from gold price trends.
- Normalization: For the LSTM model, MinMax scaling was applied to all features to standardize the values between 0 and 1. This scaling step, specifically important for neural networks, helped accelerate convergence.
- Time-Series Transformation for LSTM: Sequential LSTM modeling required transforming data into sequences. We chose a 60-day time window to provide a strong basis for the LSTM model, capturing recent trends over the past two months to forecast the next day's price.

0.10.2 Random Forest Model

Gold price prediction is influenced by many features, such as historical prices, treasury yields, economic indicators like GDP, inflation, and technical financial indicators such as moving averages and Bollinger Bands. These variables are not necessarily linear, meaning that more complex models are needed to capture the patterns in the data. This is where Random Forest shines.

0.10.2.1 Role of Random Forest:

- Handling Complex Interactions Between Features: In our project, we engineered a variety of financial indicators such as moving averages (MA50, MA200, etc.), Bollinger Bands, and relative strength index (RSI). Random Forest helps by capturing the nonlinear relationships between these indicators and the gold price. For instance, it can determine how the interaction between long-term and short-term moving averages affects future prices, or how gold prices react to changes in treasury yields.

- Reducing Overfitting: Financial markets, including gold prices, can be noisy, with unpredictable fluctuations. The Random Forest model is particularly good at reducing overfitting, thanks to its bootstrapping technique and aggregation of multiple decision trees. By training on random subsets of data, it avoids being overly sensitive to specific quirks or noise in the dataset. This is particularly important when predicting volatile markets like gold.

- Dealing with High Dimensionality: Our dataset contains numerous features, including historical gold prices, treasury yields, and various technical indicators. Random Forest is robust in handling this high-dimensional data and selecting the most important features. It identifies which indicators (e.g., moving averages or interest rate changes) have the most predictive power for future gold prices.

- Model Evaluation: In the project, we split the data into training and test sets to assess the model's performance. The Random Forest model showed strong predictive capabilities by capturing the underlying trends in gold price movement, reflected in a solid R2 Score. The model was able to effectively predict the test data based on historical trends and other indicators, providing a reliable forecast for the next day's gold price.

0.10.3 LSTM

Gold prices are highly dependent on historical trends, meaning that the order and time dependencies of past prices can heavily influence future predictions. LSTMs are well-suited to this kind of problem because they are designed to learn from sequential data, retaining long-term dependencies over time.

0.10.3.1 Role of LSTM:

- Capturing Time-Series Dependencies: In our project, gold prices are influenced by patterns over time, and LSTM is designed to model exactly that. By feeding it sequences of past gold prices (and possibly other features like treasury yields or inflation data), the LSTM can learn the long-term dependencies and recognize how past market trends influence future price movements.

- Handling Temporal Patterns: Gold prices often follow seasonal or cyclical patterns due to economic factors, geopolitical events, and market sentiment. LSTM's ability to handle time-dependent data helps capture these cyclical behaviors. For example, LSTM can track how a surge in gold demand during economic downturns or inflationary periods might affect future prices.

- Feature Engineering with LSTM: For our LSTM model, the engineered features such as moving averages, Bollinger Bands, and treasury yields would be fed in as part of a time series. LSTM processes this data sequentially, considering how each feature evolves over time. For example, if we track the 50-day moving average, the LSTM model can learn how changes in that moving average over time correlate with future price shifts.

- Predicting Future Prices: In our project, the goal was to predict gold prices for the next day or further ahead. By training the LSTM on historical sequences of prices, it learns to predict future gold prices based on the trends from the previous days. The model's internal memory helps it "remember" key patterns, such as sudden price drops or gradual upward trends, and apply this knowledge to make informed predictions.
- Challenges with LSTM: While LSTM is powerful for time-series data, it requires careful tuning of hyperparameters like sequence length, learning rate, and number of layers. In our case, we would have had to adjust these parameters to balance the model's ability to capture long-term dependencies without overfitting to short-term noise in gold prices.

0.11 Results and Discussion

```
[*********************100%%**********************]  1 of 1 completed
Gold Price on November 8, 2024: 2691.699951171875
```

This section explores the outcomes of the Random Forest and LSTM models and compares their accuracy and interpretability.

- Random Forest Model Results
  - Feature Importance Analysis: The Random Forest model provided a ranked list of feature importances, offering insights into which economic indicators and historical prices most influence gold prices. Notably, the 10-Year Treasury Yield and inflation rates were among the top predictors, suggesting that macroeconomic trends significantly impact gold prices. This analysis aligns with economic theory, where treasury yields reflect inflation expectations and real interest rates, both of which have inverse relationships with gold demand.

```
[*********************100%%**********************]  1 of 1 completed
Model R2 Score: 0.9173063936792155
Predicted Gold Price for November 8th: 2701.8280786132814
C:\Users\Nirav Shah\anaconda3\Lib\site-packages\sklearn\base.py:439: UserWarning: X does not have valid feature names, but R
andomForestRegressor was fitted with feature names
  warnings.warn(
```

- LSTM Model Results
  - Training Complexity: The LSTM model required longer training times due to the complexity of sequential data processing, but this was mitigated by using only one LSTM layer, reducing overfitting risks.

```
Merged data shape: (2504, 13)
Epoch 1/20
62/62 [==============================] - 3s 25ms/step - loss: 0.0144
Epoch 2/20
62/62 [==============================] - 1s 24ms/step - loss: 4.5010e-04
Epoch 3/20
62/62 [==============================] - 1s 24ms/step - loss: 3.4248e-04
Epoch 4/20
62/62 [==============================] - 1s 24ms/step - loss: 3.2858e-04
Epoch 5/20
62/62 [==============================] - 1s 24ms/step - loss: 3.2309e-04
Epoch 6/20
62/62 [==============================] - 1s 24ms/step - loss: 3.1888e-04
Epoch 7/20
62/62 [==============================] - 1s 24ms/step - loss: 3.0600e-04
Epoch 8/20
62/62 [==============================] - 2s 24ms/step - loss: 3.0402e-04
Epoch 9/20
62/62 [==============================] - 2s 25ms/step - loss: 3.0751e-04
Epoch 10/20
62/62 [==============================] - 2s 28ms/step - loss: 2.9749e-04
Epoch 11/20
62/62 [==============================] - 2s 31ms/step - loss: 2.7853e-04
Epoch 12/20
62/62 [==============================] - 2s 30ms/step - loss: 3.3796e-04
Epoch 13/20
62/62 [==============================] - 2s 29ms/step - loss: 2.7257e-04
Epoch 14/20
62/62 [==============================] - 2s 27ms/step - loss: 3.0806e-04
Epoch 15/20
62/62 [==============================] - 2s 27ms/step - loss: 2.9523e-04
Epoch 16/20
62/62 [==============================] - 2s 27ms/step - loss: 2.6985e-04
Epoch 17/20
62/62 [==============================] - 2s 27ms/step - loss: 3.0958e-04
Epoch 18/20
62/62 [==============================] - 2s 27ms/step - loss: 2.5183e-04
Epoch 19/20
62/62 [==============================] - 2s 26ms/step - loss: 2.5617e-04
Epoch 20/20
62/62 [==============================] - 2s 26ms/step - loss: 2.4825e-04
16/16 [==============================] - 1s 9ms/step
1/1 [==============================] - 0s 28ms/step
Predicted Next Day Gold Price: 2710.985107421875
```

0.11.1 Comparative Analysis

While both models performed competitively, the Random Forest Regressor showed an edge in predictive accuracy due to its strength in handling sequential dependencies. However, the Random Forest model offered interpretability benefits, which are critical in financial forecasting for understanding the impact of economic variables. This trade-off between accuracy and interpretability is significant, as it highlights the strengths and limitations of each approach for financial prediction.

0.12 Conclusion

In our analysis of gold price prediction, the Random Forest model provided a closer approximation to the actual price of 2691.70 on November 8 compared to the LSTM model. The Random Forest model's prediction of 2701.83 was more accurate than the LSTM's 2710.99, suggesting that Random Forest better captured the relationships between economic factors and gold prices in our dataset. This result indicates that, for our data and selected features, the Random Forest model is a more effective tool for short-term gold price prediction, offering analysts and investors a reliable method to anticipate market movements with greater accuracy.

- Economic Indicators Are Crucial: Both models identified Treasury yields and inflation as significant factors, underscoring the importance of macroeconomic trends in gold price forecasting.

- Trade-Offs Exist Between Interpretability and Accuracy: While the Random Forest model demonstrated greater accuracy in gold price prediction, it comes at the cost of reduced interpretability compared to simpler models. Although Random Forest effectively handles high-dimensional data and identifies influential economic factors, its complex structure makes it less transparent. In contrast, the LSTM model, although interpretable in terms of sequential data processing, produced slightly less accurate results. This trade-off highlights the challenge in financial forecasting: balancing the need for accuracy with the clarity needed for informed decision-making.

- This webpage displays the predicted gold price for the next day, powered by a machine learning model running in the background. Built using Streamlit, it provides an easy-to-use interface for viewing forecasted prices based on key economic factors.

0.13 Future Work

Future work could involve creating hybrid models that merge the clear interpretability of Random Forest with the ability of LSTM to recognize patterns over time. By combining these methods, we can capture complex, time-based relationships while also understanding which economic factors impact gold prices the most. Additional refinements might include using more financial indicators—like exchange rates, oil prices, or central bank policies—and incorporating sentiment analysis from financial news and social media. This approach could improve the accuracy and stability of predictions, helping investors and policymakers make more confident, well-informed sdecisions amidst economic fluctuations.

0.14 References

- Breiman, L. & Statistics Department, University of California. (2001). Random forests. In *Random Forests* (p. 4). https://www.stat.berkeley.edu/~breiman/randomforest2001.pdf
- Tang, Q., Fan, T., Shi, R., Huang, J., & Ma, Y. (2021, March 5). *Prediction of financial time series using LSTM and data denoising methods*. arXiv.org. https://arxiv.org/abs/2103.03505
- Davis, J. (2024, September 5). *How to build LSTM models for time series prediction in Python*. Statology. https://www.statology.org/how-to-build-lstm-models-for-time-series-prediction-in-python/
- *The future trajectory of gold prices (2024-2026): navigating economic and geopolitical crossroads*. (2023, December 28). Kitco News. https://www.kitco.com/opinion/2023-12-28/future-trajectory-gold-prices-2024-2026-navigating-economic-and-geopolitical
- Baur, D. G., & Lucey, B. M. (2009). Is Gold a Hedge or a Safe Haven? an Analysis of Stocks, Bonds and Gold. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.952289
- Zhang, C., Sjarif, N. N. A., & Ibrahim, R. (2023b). Deep learning models for price forecasting of financial time series: A review of recent advancements: 2020–2022. *Wiley Interdisciplinary Reviews Data Mining and Knowledge Discovery*, *14*(1). https://doi.org/10.1002/widm.1519
- https://finance.yahoo.com/quote/GC%3DF/
- https://fred.stlouisfed.org/series/T10YIE
- https://data.worldbank.org/indicator/NY.GDP.MKTP.CD
- https://finance.yahoo.com/markets/bonds/