Problem Definition & Assumptions

Problem Definition:

The goal of this project is to forecast the Federal Funds Effective Rate, commonly referred to as the interest rate, for January 2025. This involves creating a predictive model that relies on historical interest rate data, with the option to incorporate other relevant economic indicators. The emphasis of this task lies in the interpretability of the model and the rationale behind its design, rather than achieving the highest prediction accuracy.

Scope of Prediction:

The prediction focuses on a one-month horizon, specifically for January 2025. Interest rates signify the cost of borrowing money as a percentage. This serves as a crucial metric that impacts financial markets, lending activities, and overall economic growth.

Data Description:

The primary dataset utilized is the Federal Funds Effective Rate data, sourced from the Federal Reserve Economic Data (FRED). This dataset includes monthly interest rate figures that illustrate historical trends, enabling the creation of a forecasting model. Supplementary datasets, such as employment figures, inflation data, or GDP trends, may be incorporated if they demonstrate meaningful correlations with interest rates.

Assumptions:

1. Stationarity of Interest Rates:
   It is assumed that interest rates exhibit stationarity, meaning their statistical characteristics, such as mean and variance, remain stable over time. This assumption simplifies the modeling process and aligns with established methods in time-series analysis.
2. Consistency of Economic Context:
   The macroeconomic factors influencing interest rates historically are assumed to continue operating similarly up to January 2025. The model does not account for unpredictable events or major policy shifts.
3. Data Reliability and Completeness:
   The dataset used for model training is presumed to be accurate, complete, and free from significant gaps or anomalies. This ensures that the model is based on dependable data, leading to credible predictions.
4. Validity of Historical Relationships:
   The model assumes that the relationships identified in historical data remain applicable for future predictions. While this assumption is reasonable for short-term forecasting, its reliability decreases for longer time frames.

5. Correlation Between Features and Target:
   When additional data is used, it is assumed to have a substantial and consistent relationship with the Federal Funds Rate. For instance, inflation trends are often linked to changes in interest rates, where higher inflation may indicate an upward adjustment in rates.
6. Randomness in Interest Rates:
   Interest rates are considered a random variable, potentially following patterns such as normal distributions or autoregressive processes, while also being influenced by global and policy-driven constraints.
7. Economic Volatility:
   Predictions rely heavily on economic data such as GDP growth, unemployment rates, and inflation indicators. However, economic conditions can change rapidly, leading to revisions in forecasts. Unforeseen events, such as geopolitical tensions or natural disasters, can also disrupt predictions.

Model Selection

The model chosen for this task is a **Long Short-Term Memory (LSTM) neural network**, which is a specialized architecture within Recurrent Neural Networks (RNNs). LSTMs are particularly suited for time-series forecasting tasks due to their ability to capture long-term dependencies and temporal patterns in sequential data. Here are the primary considerations and justifications for selecting the LSTM model:

Nature of the Data

- The dataset consists of **monthly Federal Reserve interest rates**, which exhibit temporal dependencies and trends over time.
- Traditional machine learning models, such as linear regression or random forest, often struggle to incorporate sequential patterns and temporal correlations.
- LSTMs excel in processing sequential data due to their gated architecture, which can selectively retain or forget information as needed.

Capturing Temporal Dependencies

Time-series data often have trends, seasonality, and irregular fluctuations. The model, through mechanisms like input, forget, and output gates, effectively capture both short-term and long-term patterns. This allows for more accurate forecasting.

Scalability and Flexibility.

The model's ability to handle varying sequence lengths makes it robust for datasets with missing or irregular data points. The model can also be easily scaled or adapted to incorporate additional features if needed

Implementation Details

- **Two LSTM Layers**: The first layer processes the sequential data and passes patterns to the second layer. This enhances the feature representation.
- **Dropout Layers**: This adds regularization to the model
- **Dense Output Layer**: Ensures the output is a single scalar value representing the predicted interest rate.

Empirical Justification

In practice, LSTMs have demonstrated superior performance in various financial and economic time-series forecasting tasks, including stock prices, interest rates, and inflation.

Model Architecture

Input

The input to the model consists of sequences of interest rates, where each sequence contains data from the past 12 months. This means the input dimension is defined as:

- **Number of sequences**: The total number of training samples generated from the dataset, which depends on its length.
- **Sequence length**: 12 months of historical data.
- **Features per timestep**: 1 (the Federal Funds rate).

For example, if the data has 100 rows, the first input sequence will include rates from months 1–12, and the corresponding target is the rate for month 13. This sliding window approach helps the model learn temporal dependencies.

Layers and Their Functions

1. **First LSTM Layer**:
   - Purpose: This layer captures short-term dependencies and sequential patterns in the input data. LSTM cells have a memory mechanism (cell state) and gating mechanisms (input, output, and forget gates) that allow them to retain or discard information over time.
   - Mechanics: Each LSTM cell processes one time step at a time, updating its hidden state and cell state based on the input for that timestep and the previous state. The output of this layer is a sequence of hidden states, one for each timestep, which is passed to the next layer.

2. **Second LSTM Layer**:
   - Purpose: This layer refines the representation learned by the first LSTM layer, capturing higher-order temporal patterns. It helps the model learn more complex dependencies over time.
   - Mechanics: This layer also processes sequences but aggregates information from the outputs of the first LSTM layer. Since the `return_sequences` parameter is set to `False`, it outputs only the final hidden state, summarizing the entire sequence.

3. **Dense Layer**:
- Purpose: The dense layer acts as the final prediction layer. It transforms the single hidden state from the second LSTM layer into a scalar output, which represents the predicted interest rate for the next month.
- Mechanics: A linear activation function is used, as this is a regression problem. The weights in this layer are learned to map the features extracted by the LSTM layers to the predicted value.

Training Strategy

The training process is designed to ensure the model learns to predict the Federal Funds rate effectively by capturing both short-term and long-term temporal dependencies in the data. The steps are as follows:

1. Data Processing
    a. The raw dataset is first normalized using MinMaxScaler to scale the Federal Funds rates to a range between 0 and 1. Normalization ensures that the data magnitude does not dominate during training and accelerates convergence.
    b. Input sequences are created using a sliding window approach, where each sequence consists of 12 months of historical interest rates, and the corresponding target is the rate for the 13th month.

2. Train-Validation-Test Split
    a. Walk Forward validation - To mimic real-world scenarios, a walk-forward validation strategy is employed. The model is initially trained on a fixed training window, validated on a subsequent segment, and the process is repeated by moving the window forward by a fixed step size. This ensures the model is evaluated on unseen, temporally ordered data, providing a realistic measure of its predictive performance.

3. Loss Function
    a. The **mean squared error (MSE)** is used as the loss function, as it effectively penalizes large prediction errors and is suitable for regression tasks. During training, the model minimizes this loss, encouraging it to align predictions closely with the true Federal Funds rates.

4. Training Configuration
    a. The model is trained using the **Adam optimizer**, known for its adaptive learning rate and efficient convergence. A learning rate of 0.001 is chosen to balance learning speed and stability.
    b. Training is conducted in mini-batches, with a batch size of 32. Mini-batching allows for efficient gradient updates and memory utilization.
    c. The model is trained for a maximum of 10 epochs for each walk forward step.

Temporal Relationships and Data Flow

- The first LSTM layer focuses on local patterns within the input sequence (e.g., identifying trends or cycles over shorter periods).
- The second LSTM layer aggregates and refines this information, capturing global dependencies over the entire sequence.
- The dense layer translates the summarized information from the sequence into the final output.

Why This Architecture Works

1. **Sequential Data Handling**: LSTMs are specifically designed to handle sequential data, making them ideal for learning patterns in time series like Federal Funds rates.
2. **Memory Mechanism**: By maintaining a cell state, LSTMs retain relevant information over long sequences, which is crucial for capturing both short-term trends and long-term dependencies.
3. **Non-linear Modeling**: The architecture can model the non-linear relationships between past and future rates, which are typical in economic data.

This architecture is well-suited for capturing the dynamics of interest rates while remaining computationally efficient.

Evaluation Metrics

In evaluating the performance of our model for predicting Federal Reserve interest rates, we selected metrics that align with the forecasting of continuous numerical values. The following metrics were used:

1. Mean Squared Error (MSE):
- Why Chosen: MSE measures the average squared difference between the predicted and actual values, penalizing larger errors more heavily. It provides a straightforward and interpretable measure of prediction accuracy.
- Interpretation: A lower MSE indicates better model performance, as it signifies smaller deviations from the actual interest rates.

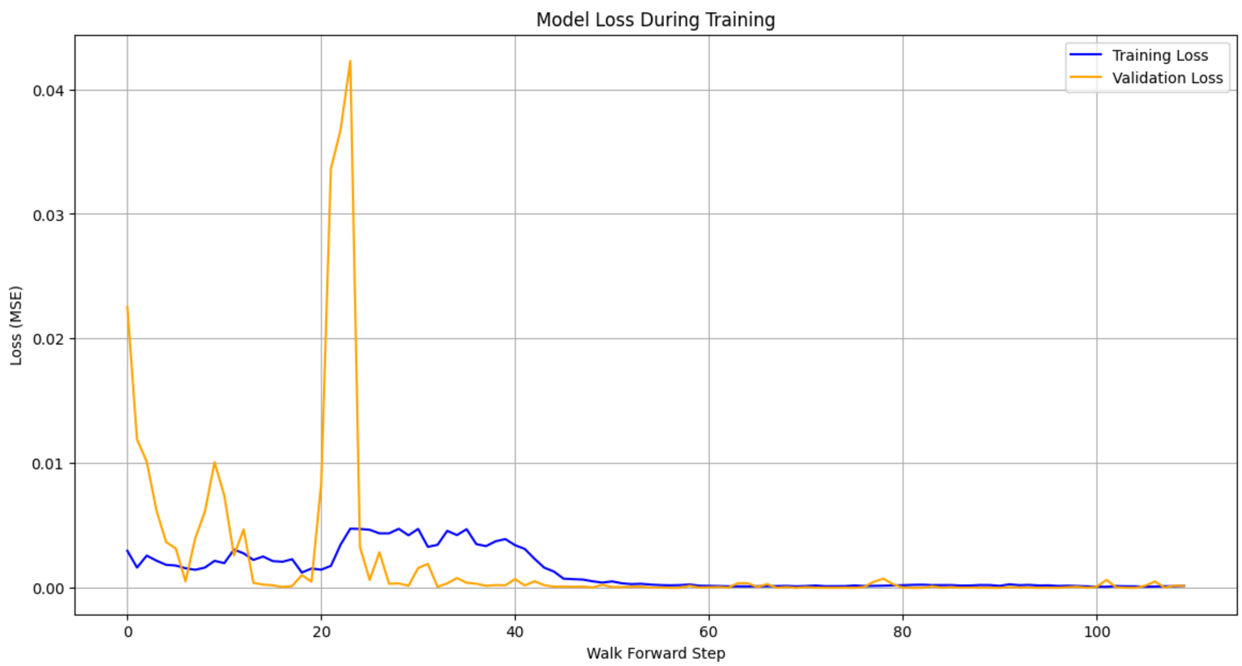
2. Root Mean Squared Error (RMSE):
- Why Chosen: RMSE is the square root of MSE and is expressed in the same units as the target variable (interest rates). This makes it easier to relate the error magnitude to the actual predictions.
- Interpretation: RMSE provides an intuitive sense of the typical size of the prediction error. For example, if the RMSE is 0.05, we can expect the model's predictions to be off by ~0.05% on average.

3. Mean Absolute Error (MAE):
- Why Chosen: MAE calculates the average magnitude of errors in a set of predictions without considering their direction (i.e. underprediction or overprediction). Unlike MSE, it does not overly penalize large errors.
- Interpretation: MAE gives a clear understanding of the average prediction error. For instance, an MAE of 0.03 indicates the model's predictions are on average 0.03% off.

4. Validation Loss (during training):
- Why Chosen: Validation loss, calculated as MSE on the validation set, was monitored during training to identify overfitting or underfitting. By comparing training loss with validation loss, we ensure that the model generalizes well to unseen data.
- Interpretation: A significant gap between training loss and validation loss suggests overfitting, while similar values indicate a well-generalized model.

5. Visualization of Predicted vs. Actual Values:
- Why Chosen: Beyond numerical metrics, visual comparison of the model's predictions against actual interest rates over time helps to understand patterns and discrepancies.
- Interpretation: A close alignment between the predicted and actual curves indicates good model performance. Deviations can point to areas for further model improvement.

Results and Interpretation



The training and validation loss fluctuates for the first few iterations of the walk forward training but as the training progresses, our model learns to generalize predict interest rates with better accuracy. Thereby highlighting the effectiveness of the walk forward validation with a seemingly simple LSTM model.