

Comprehensive Stock Analysis and Prediction Task

Objective: As a Data Scientist, your task is to develop a Python program that takes a CSV file containing historical stock data as input, conducts a comprehensive analysis of the stock, and predicts its future performance.

Requirements:

1. Data Validation and Cleaning:

- Read the input CSV file provided by the user and return the predictions.
- Validate the structure and format of the data.
- Handle missing values, outliers, and any inconsistencies in the data.
- Ensure data integrity and accuracy.

2. Stock Analysis:

- Calculate basic statistical measures such as mean, median, standard deviation, etc., for each column (Date, Open, High, Low, Close, Adj Close, Volume).
- Visualize the historical trends of the stock using appropriate plots (e.g., line plots, candlestick plots).
- Conduct exploratory data analysis to identify patterns, trends, and anomalies in the data.
- Compute correlations between different variables and assess their significance.

3. Predictive Modeling:

- Utilize machine learning techniques (e.g., linear regression, ARIMA, LSTM) to build a predictive model for the stock.
- Split the data into training and testing sets.
- Determine if there are any seasonal patterns or cyclical trends in the stock's behavior.
- Apply time series decomposition techniques to isolate seasonal, trend, and residual components.
- Train the model using historical data.
- Evaluate the model's performance using appropriate metrics (e.g., RMSE, MAE).
- Generate predictions for future stock prices based on the trained model.

4. Deployment and Monitoring:

- Develop a plan for deploying the predictive model into production.
- Create monitoring mechanisms to track the performance of the deployed model.

Deliverables:

- Python script or Jupyter notebook for data validation, analysis, and prediction.
- Thoroughly documented with comments.

- Plots/charts showing historical trends and predictive performance.
- Deployment plan and monitoring framework for the predictive model.

All deliverables pushed to a GitHub repo. Share repository link for access and review.

Additional Information:

You may leverage external libraries such as pandas, NumPy, matplotlib, scikit-learn, and TensorFlow/PyTorch for data manipulation, analysis, and modeling. Ensure that the code is well-structured, modular, and scalable to handle different stocks and time periods. Good luck!