**FAANG-tastic Insights: Predict Stock Prices with Regression and MLflow**

**1. Project Overview**

This project aims to develop a stock price prediction model for FAANG stocks using regression techniques and deploy it as a user-friendly Streamlit web application. The model helps financial analysts and investors make data-driven decisions by forecasting stock prices.

**2. Explanation of Models and Methodologies**

**2.1 Data Preprocessing**

* **Handling Missing Values**: Rows with extensive missing data are removed.Median imputation is applied where necessary.
* **Date Conversion**: The 'Date' column is converted to a datetime format for time-series analysis.
* **Feature Engineering**: Selecting relevant features ('Open', 'High', 'Low', 'Volume') to predict the target variable ('Close').
* **Data Normalization**: Min-Max scaling is used to standardize numerical features.

**2.2 Machine Learning Models**

**Linear Regression**

* Assumes a linear relationship between stock features and closing prices.
* Suitable for general trend prediction but lacks flexibility in capturing non-linear trends.

**Random Forest Regressor**

* An ensemble learning method using multiple decision trees.
* Handles non-linearity and captures complex patterns in stock prices.
* Provides better predictive performance compared to linear regression.

**2.3 Model Evaluation Metrics**

* **Mean Absolute Error (MAE)**: Measures the average magnitude of errors.
* **Root Mean Squared Error (RMSE)**: Emphasizes larger errors.
* **R-squared (R²) Score**: Indicates how well the model explains variance in stock prices.

**2.4 MLflow Integration**

* MLflow is used for **experiment tracking**, logging metrics (MAE, RMSE, R²), and storing models.
* Enables comparison of different models to select the best one for deployment.

**3. Instructions for Running and Deploying the App**

**3.1 Running Locally**

1. Install required dependencies:
2. pip install pandas numpy scikit-learn mlflow streamlit matplotlib seaborn
3. Start the MLflow tracking server:
4. mlflow ui
5. Run the Streamlit application:
6. streamlit run app.py

**3.2 Deployment on Cloud (AWS/Streamlit Sharing)**

1. **Streamlit Sharing:**
   * Push the project to GitHub.
   * Use share.streamlit.io for direct hosting.

**4. Presentation: Summary of Results and Business Insights**

**4.1 Key Results**

* The **Random Forest model outperformed Linear Regression**, providing lower MAE and RMSE values.
* MLflow logs showed **better accuracy trends with hyperparameter tuning**.
* The **Streamlit app provides an interactive and easy-to-use interface** for making predictions based on user input.

**4.2 Business Insights**

* **Investment Strategy**: Investors can use the model to analyze FAANG stock trends and improve decision-making.
* **Portfolio Analysis**: Helps in identifying high-performing stocks for diversification.
* **Trading Insights**: Day traders can use short-term predictions to optimize entry/exit points.

**5. Conclusion**

This project demonstrates the power of machine learning and MLflow in stock market prediction. With **automated model tracking and real-time predictions**, users can make informed financial decisions with confidence.

**6. Future Enhancements**

* Incorporating **LSTM (Long Short-Term Memory) models** for better time-series forecasting.
* Adding **sentiment analysis** using financial news data to improve predictions.
* Enhancing the **app UI with additional stock indicators** like RSI and MACD for technical analysis.

**7. References**

* MLflow Documentation: <https://mlflow.org/>
* Streamlit Documentation: <https://docs.streamlit.io/>
* Scikit-Learn: <https://scikit-learn.org/>