Data Analysis Project 1.

Predicting Stock Returns Using Machine Learning

1. Project Overview

The objective of this project is to develop a machine learning-based model that can predict stock returns based on historical data and other relevant financial indicators. The model should provide insights into future stock price movements, helping investors make informed decisions.

2. Grading and Submission:

The grading of the project will be divided into two parts: Group-level and Individual-level works.

At the group-level:

- 10% credits will be granted based on the completion and degree of perfection.
- Submit the program (code) used for data collection and preprocessing.
- Submit a short summary of data collection and preprocessing.
- Submit any other thing that you are comfortable to share at the group-level.

At the individual-level:

- 10% credits will be granted based on the completion and machine learning performance (i.e., completion rate*F1 score).
- Submit the program (code) used for individual analysis (anything that your group did not share).
- Submit a short summary of (1) additional data sources that are not shared by the group (2) descriptions of feature selection (3) descriptions of model selection (4) report your model performance (5) economic interpretation on important features and its effects on stock return.

3. Project Goals

- **Develop Predictive Models:** Create machine learning models to predict short-term and long-term stock returns.
- **Feature Engineering:** Identify and select relevant features that impact stock returns, such as historical prices, trading volume, macroeconomic indicators, and company-specific factors.
- **Backtesting and Validation:** Implement rigorous backtesting to evaluate model performance over historical data and validate its accuracy on out-of-sample data.
- **Deployment:** Deploy the model as an interactive tool or API for real-time predictions.

4. Functional Requirements

4.1 Data Collection

- **Source:** Collect historical stock price data from reliable financial data providers such as Yahoo Finance, CRSP, or CSMAR other financial databases.
- **Data Types:**
  - Historical Stock Prices: Open, high, low, close, adjusted close, and volume.
  - Technical Indicators: Moving averages, RSI, MACD, etc.
  - Fundamental Data: Earnings, dividends, P/E ratio, etc.
  - Macroeconomic Indicators: Interest rates, GDP growth, inflation, etc.
- **Data Frequency:** Daily, weekly, and monthly data, depending on the prediction horizon.

3.2 Data Preprocessing

- **Data Cleaning:** Handle missing data, outliers, and anomalies.
- **Normalization/Standardization:** Apply appropriate scaling to ensure model performance.
- **Feature Engineering:** Create additional features such as lagged returns, volatility measures, and sentiment analysis from news data.

3.3 Model Development

- **Model Selection:** Evaluate different machine learning models, including but not limited to:
  - **Linear Models:** Linear Regression, Lasso, Ridge
  - **Tree-based Models:** Random Forest, Gradient Boosting, XGBoost
  - **Neural Networks:** Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM)
  - **Support Vector Machines (SVM)**
- **Training:** Train models using historical data with a focus on minimizing overfitting and improving generalization.
- **Hyperparameter Tuning:** Use techniques like grid search or Bayesian optimization to find optimal parameters.

3.4 Model Evaluation

- **Performance Metrics:** Evaluate model performance using metrics such as:
  - **Mean Squared Error (MSE)**
  - **Mean Absolute Error (MAE)**
  - **R-squared (R^2)**
  - **Sharpe Ratio**
- **Backtesting:** Simulate model performance on unseen historical data to assess predictive power and potential profitability.
- **Cross-Validation:** Implement k-fold cross-validation to ensure model robustness.

6. Resources Required

- **Tools:**
  - Python, TensorFlow/PyTorch, Scikit-learn, Pandas, SQL databases, Cloud services (AWS/Azure/GCP).
  - Most open resource references might use python code. I would suggest you use Python. Feel free to search on StackFlow.com for the basic set up.
  - Google colab is a highly suggested platform for python programming and data storage. https://colab.research.google.com/
- **References**:
  - Gu, S., Kelly, B., & Xiu, D. (2020). Empirical asset pricing via machine learning. The Review of Financial Studies, 33(5), 2223-2273.
  - Leippold, M., Wang, Q., & Zhou, W. (2022). Machine learning in the Chinese stock market. Journal of Financial Economics, 145(2), 64-82.
- **Weblink tutorials:**
  - https://www.geeksforgeeks.org/stock-price-prediction-using-machine-learning-in-python/
  - You can find many other helpful resources online. Feel free to use them as your guidance.

7. Success Criteria

- **Accuracy:** The model achieves an acceptable level of prediction accuracy (e.g., R^2, F1 score).
- **Usability:** Your model should be user-friendly and provide actionable insights for investors.
- **Performance:** The model is capable of providing predictions within the specified time frame and handling the expected data volume.