Project Report

Title: Stock Price Prediction using Machine Learning
Internship: Coding Samurai
Prepared by: Henil Parag Patel
Date: 10-08-2025

1. Abstract

This project aims to predict future stock prices using machine learning techniques based on historical data. The study focuses on two approaches: a **Linear Regression** model for a baseline prediction and a **Multi-Layer Perceptron (MLP) Neural Network** for capturing non-linear trends. The models were trained on historical stock data collected using the Yahoo Finance API (`yfinance`). The project covers **data preprocessing, exploratory data analysis (EDA), model training, evaluation, and forecasting**. The results indicate that the neural network model provides better adaptability to market fluctuations, while the linear regression model is simpler and easier to interpret.

2. Introduction

Stock market prediction is one of the most challenging and intriguing problems in finance. Prices are influenced by numerous factors such as economic indicators, investor sentiment, and market news. However, historical price data can still reveal patterns useful for forecasting.
The goal of this project is to design a predictive system that:

1. Analyzes past stock prices.
2. Trains machine learning models.
3. Forecasts future prices for short-term decision-making.

3. Objectives

- To collect and preprocess historical stock price data.
- To perform exploratory data analysis (EDA) to understand trends and patterns.
- To implement **Linear Regression** as a baseline model.
- To implement an **MLP Neural Network** for enhanced prediction accuracy.
- To evaluate models using MAE, RMSE, and R^2 metrics.
- To forecast stock prices for the next 30 days.

4. Literature Review

Stock price prediction has been studied using statistical methods, machine learning, and deep learning. Traditional regression models are interpretable but may fail to capture complex market patterns. Neural networks, on the other hand, are powerful in learning non-linear dependencies but require careful tuning. Previous works have applied **ARIMA models, LSTM networks, and ensemble methods**, each with varying degrees of success. This project uses **Linear Regression** and **MLP Neural Networks** for simplicity and effectiveness.

5. Methodology

5.1 Data Collection

- Source: Yahoo Finance (`yfinance` library).
- Stock Symbol: *[Example: AAPL – Apple Inc.]*
- Period: *[Example: 5 years of daily data]*.
- Attributes used: Date, Open, High, Low, Close, Volume.

5.2 Data Preprocessing

- Removed missing values.
- Sorted data by date.
- Scaled features using `MinMaxScaler` (for neural network).

5.3 Exploratory Data Analysis (EDA)

- Plotted historical stock prices.
- Calculated moving averages (50-day, 200-day).
- Observed volatility and trend patterns.

5.4 Model Development

a) Linear Regression Model:

- Features: Previous day’s price(s).
- Target: Next day’s closing price.
- Implemented using `scikit-learn`.

b) Neural Network Model (MLP):

- Layers: Input → Hidden (128 neurons) → Hidden (64 neurons) → Output.
- Activation Functions: ReLU (hidden), Linear (output).
- Loss Function: Mean Squared Error.
- Optimizer: Adam.

5.5 Model Evaluation

- Metrics: Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), R^2 Score.
- Train-Test Split: 80%-20%.

5.6 Future Forecasting

- Predicted stock prices for the next 30 days using recursive predictions.

6. Results

Model	MAE	RMSE	R^2 Score
Linear Regression	X.XXX	X.XXX	0.XXX
Neural Network MLP	X.XXX	X.XXX	0.XXX

- The **MLP Neural Network** showed lower MAE and RMSE compared to Linear Regression.
- Forecast plots indicated that the MLP model better captured market fluctuations.

7. Discussion

- **Linear Regression** is fast, interpretable, but limited in handling complex trends.
- **Neural Network** adapts better to non-linear patterns and can produce smoother forecasts.
- Performance depends on data quality, feature selection, and hyperparameter tuning.

8. Conclusion

This project successfully demonstrated stock price forecasting using both traditional and modern ML methods. While predictions are not guaranteed due to market volatility, machine learning provides valuable insights for trend estimation. Future work may include integrating **LSTM models, sentiment analysis from financial news, and macroeconomic indicators** to improve accuracy.

9. References

- Yahoo Finance API Documentation: https://pypi.org/project/yfinance/
- Scikit-learn Documentation: https://scikit-learn.org/
- TensorFlow/Keras Documentation: https://www.tensorflow.org/

10. Appendix

- The full Python notebook (Stock Prices Prediction. ipynb) is `attached.`

- Screenshots of EDA and predictions included in project files.