Assessment Report

on

"Stock Price Prediction"

submitted as partial fulfillment for the award of

BACHELOR OF TECHNOLOGY DEGREE

SESSION 2024-25

in

CSE(AIML)

By

GROUP: 5

SHIKSHA AGRAWAL(202401100400175)

VANSHIKA ARORA(202401100400208)

SRISHTI(202401100400183)

SRISH JANA(202401100400189)

UPWAN SINGH(202401100400200)

Section: C

Under the supervision of

"Mr.Abhishek Shukla"

KIET Group of Institutions, Ghaziabad

1. Introduction

In the ever-evolving financial market, predicting stock prices remains one of the most sought-after and challenging tasks. Accurate stock price forecasting is crucial for investors, traders, and financial institutions to make informed decisions and manage risks effectively. With the availability of historical stock market data and advancements in machine learning, it is now possible to create models that can help predict future prices based on past trends and indicators. This project leverages regression analysis, a key supervised machine learning technique, to predict the **next-day closing price** of a stock using historical data. The model utilizes features such as previous closing price and moving averages to learn and forecast future values, offering insights into short-term market movements.

---

2. Problem Statement

This project aims to predict the **next-day closing price** of a stock using historical data. By applying regression techniques on features like previous closing prices and moving averages, the model seeks to provide accurate short-term forecasts to assist in informed financial decision-making.

---

3. Objectives

- Data Acquisition and Preprocessing.
- Feature Engineering and Prediction.
- Model Development using Linear Regression.
- Evaluation of model performance.
- Visualization of prediction and trends.

---

4. Methodology

- **Data Collection**: The user uploads a CSV file containing the dataset.
- **Data Preprocessing**:
    - The Date column is converted to datetime format to ensure correct time-series ordering.
    - The data is sorted chronologically to maintain sequential integrity.
    - Rows with missing values (from moving average or shifting) are removed using dropna()
- **Feature Engineering:**
    - The feature set X includes :Prev_Close ,5Day_MA
    - The target variable y is: Next_Close.

---

5. Data Preprocessing

The dataset is cleaned and prepared as follows:

- Data Conversion and Sorting.
- New Columns are generated: Prev_Close,5Day_MA,Next_Close.
- Rows with NaN values (introduced by the moving average and shifting) are removed using dropna() to ensure model training on complete data..
- The dataset is split into 80% training and 20% testing.

---

6. Model Implementation

Linear Regression is a supervised machine learning algorithm used to model the relationship between a dependent continuous variable and one or more independent variables. In this project, Linear Regression is employed to predict the next-day closing price of a stock based on historical features.

---

7. Evaluation Metrics

The following metrics are used to evaluate the model:

- Mean Absolute Error (MAE)
- **Root Mean Squared Error (RMSE)**
- R^2 Score (Coefficient of Determination)
- Visual Comparison

---

8. Results and Analysis

- The **Linear Regression model** was successfully trained and achieved a reasonable **R^2 score**.
- The **MAE and RMSE values** show that the model performs well with relatively low average prediction error.
- The **prediction plot** closely follows the actual next-day prices, especially in stable market periods, validating the model's ability to capture short-term trends.

---

9. Conclusion

In this project, a regression-based model was developed to predict the next-day closing price of a stock using historical data. By leveraging features such as the previous day's closing price and the 5-day moving average, the model was able to capture short-term price trends with reasonable accuracy.

---

10. References

- KAGGLE : https://www.kaggle.com/datasets/rohanrao/nifty50-stock-market-data

---

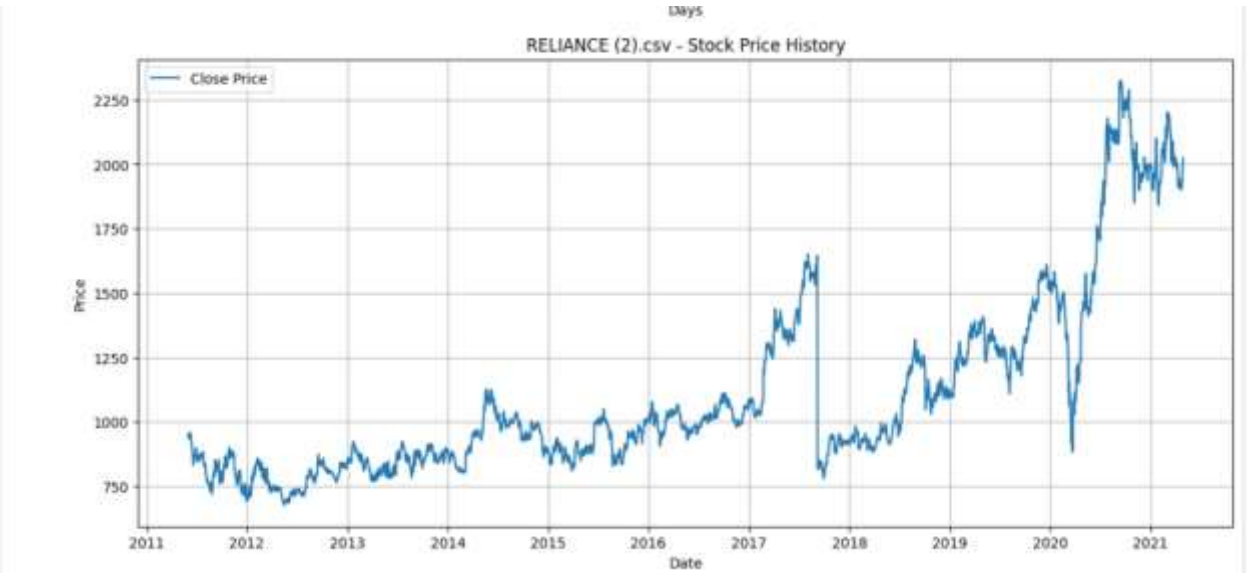
11.Code

```
# ☑ STOCK PRICE PREDICTION (NEXT-DAY) WITH MODEL ACCURACY DISPLAYED


import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

from google.colab import files

from sklearn.linear_model import LinearRegression

from sklearn.metrics import mean_absolute_error, mean_squared_error,
r2_score

from sklearn.model_selection import train_test_split


# 📤 Upload a single stock CSV file (e.g., INFY.csv)

uploaded = files.upload()
```

```
filename = list(uploaded.keys())[0]
df = pd.read_csv(filename)

# 🖌️ Preprocess
df['Date'] = pd.to_datetime(df['Date'])
df = df.sort_values('Date')
df['Next_Close'] = df['Close'].shift(-1)
df['Prev_Close'] = df['Close']
df['5day_MA'] = df['Close'].rolling(window=5).mean()
df = df.dropna()

# 🎯 Features & Target
X = df[['Prev_Close', '5day_MA']]
y = df['Next_Close']

# 🔀 Split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, shuffle=False)

# 🤖 Train
model = LinearRegression()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
```

```
# 📏 Evaluate
mae = mean_absolute_error(y_test, y_pred)
rmse = np.sqrt(mean_squared_error(y_test, y_pred))
r2 = r2_score(y_test, y_pred)

print(f"\n📊 Model Evaluation for: {filename}")
print(f"MAE   (Mean Absolute Error)       : {mae:.2f}")
print(f"RMSE  (Root Mean Squared Error)   : {rmse:.2f}")
print(f"☑️ Model Accuracy (R² Score)      : {r2:.2%}")  # Show as percentage

# 📈 Plot
plt.figure(figsize=(14, 6))
plt.plot(y_test.values, label='Actual Next-Day Price', linewidth=2)
plt.plot(y_pred, label='Predicted Next-Day Price', linewidth=2)
plt.title(f'{filename} - Next-Day Stock Price Prediction')
plt.xlabel('Days')
plt.ylabel('Price')
plt.legend()
plt.grid(True)
plt.show()
```

```
# 📉 Historical Price Trend
plt.figure(figsize=(14, 6))
plt.plot(df['Date'], df['Close'], label='Close Price')
plt.title(f'{filename} - Stock Price History')
plt.xlabel('Date')
plt.ylabel('Price')
plt.legend()
plt.grid(True)
plt.show()
```




```
Model Evaluation for: RELIANCE (2).csv
MAE   (Mean Absolute Error)       : 25.83
RMSE  (Root Mean Squared Error)   : 36.66
Model Accuracy (R² Score)         : 98.96%
```