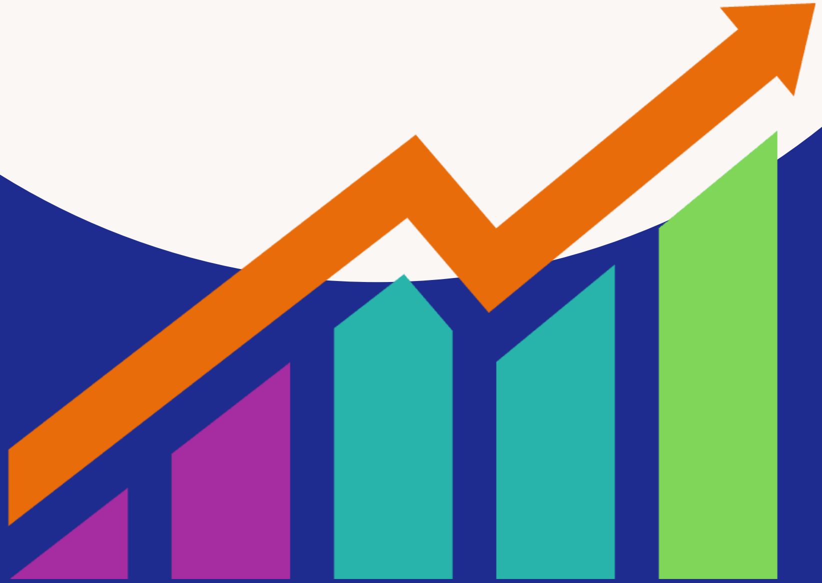
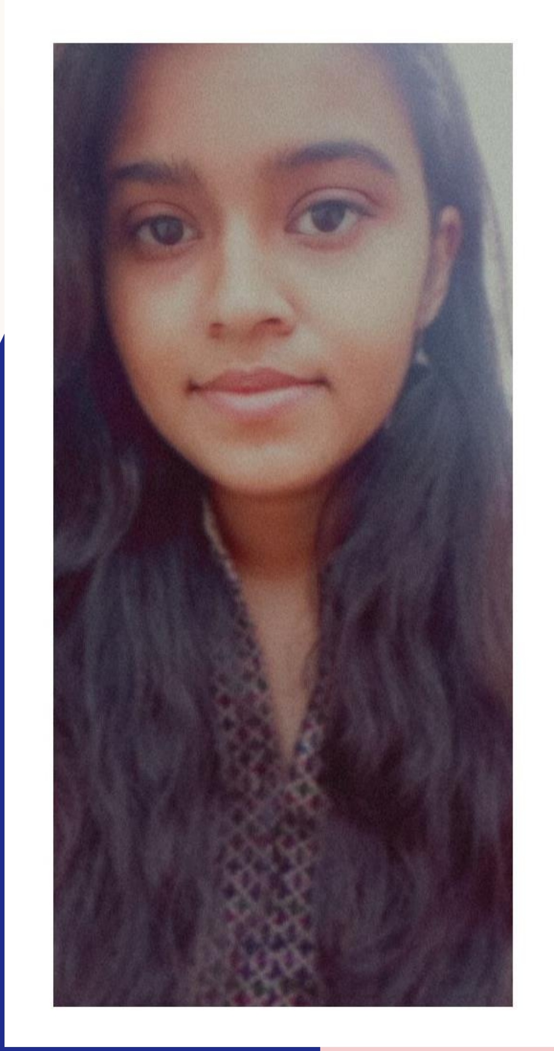
Submitted by
Harshita
MCA semester 1

STOCK PRICE PREDICTION USING PYTHON





CONTENT

Problem statement

Key challenges

Objective

Introduction

Machine learning algorithms

Data flow Diagram

Module description

Linear regression analysis

Statistical tools

Python libraries

Future enhancement

Conclusion

JUST A NORMAL DAY AT THE NATION'S MOST IMPORTANT FINANCIAL INSTITUTION...
I'VE GOT A STOCK HERE THAT COULD REALLY EXCEL
REALLY EXCEL?
EXCEL?
SELL?
SELL?
SELL
SE
SELL!
SELL!
KAL 89
SELL!
SELL!
ELL!
SELL!
THIS IS MADNESS! I CAN'T TAKE ANYMORE
GOOD BYE?
BYE?
Buy?
BUY!
Buy!
BUY!!
Bu
Buy!
Buy!
GOOD BYE!
I'VE GOT A STOCK HERE..

PROBLEM STATEMENT

The problem revolves around developing a predictive model for stock prices using linear regression. The primary objective is to create a model that forecasts the next day's closing stock price based on historical data. The challenge lies in navigating financial markets' inherent complexity and volatility, where numerous factors contribute to stock price movements.

KEY CHALLENGES

1. **Dynamic Nature of Financial Markets:**

   Financial markets are influenced by a multitude of factors, including economic indicators, geopolitical events, and market sentiment. Capturing this dynamism is crucial for accurate predictions.

2. **Limited Predictive Power of Historical Data:**

   Stock prices are affected by both historical trends and real-time developments. Constructing a predictive model that effectively balances historical patterns with current market conditions is a critical challenge.

3. **Model Simplicity vs. Accuracy Trade-off:**

   Linear regression, while a straightforward approach, may oversimplify the complex relationships inherent in stock price movements. Striking the right balance between model simplicity and predictive accuracy is a key consideration.

4. **Evaluation Metrics for Model Performance:**

   Establishing robust evaluation metrics is essential for determining the efficacy of the predictive model. Identifying suitable measures that reflect the nuances of stock price prediction is part of the challenge.

OBJECTIVE

- Develop a predictive model for stock price using historical data.
- Evaluate the model's performance and accuracy.
- Implement a user-friendly interface for users to make predictions.
- Evaluate model performance using appropriate metrics such as Mean Squared Error and R-squared.

INTRODUCTION

WHAT IS STOCK?

A stock, also known as equity, is a security that represents the ownership of a fraction of the issuing corporation. Units of stock are called "shares" which entitles the owner to a proportion of the corporation's assets and profits equal to how much stock they own.

WHAT IS STOCK MARKET?

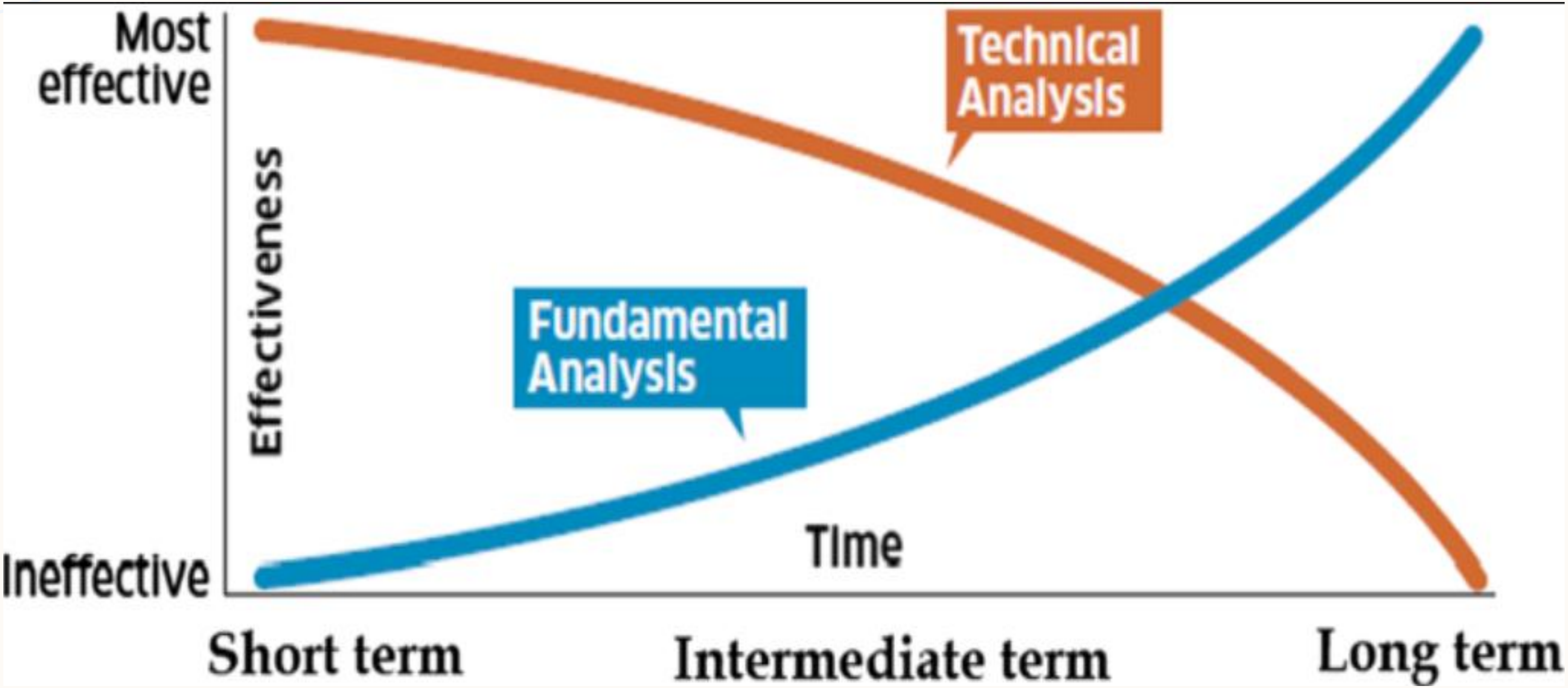
The stock market is a component of a free-market economy. It allows companies to raise money by offering stock shares and corporate bonds. It allows investors to participate in the financial achievements of the companies, make profits through capital gains, and earn income through dividends.

WHY ARE PREDICTING STOCK PRICES IMPORTANT?

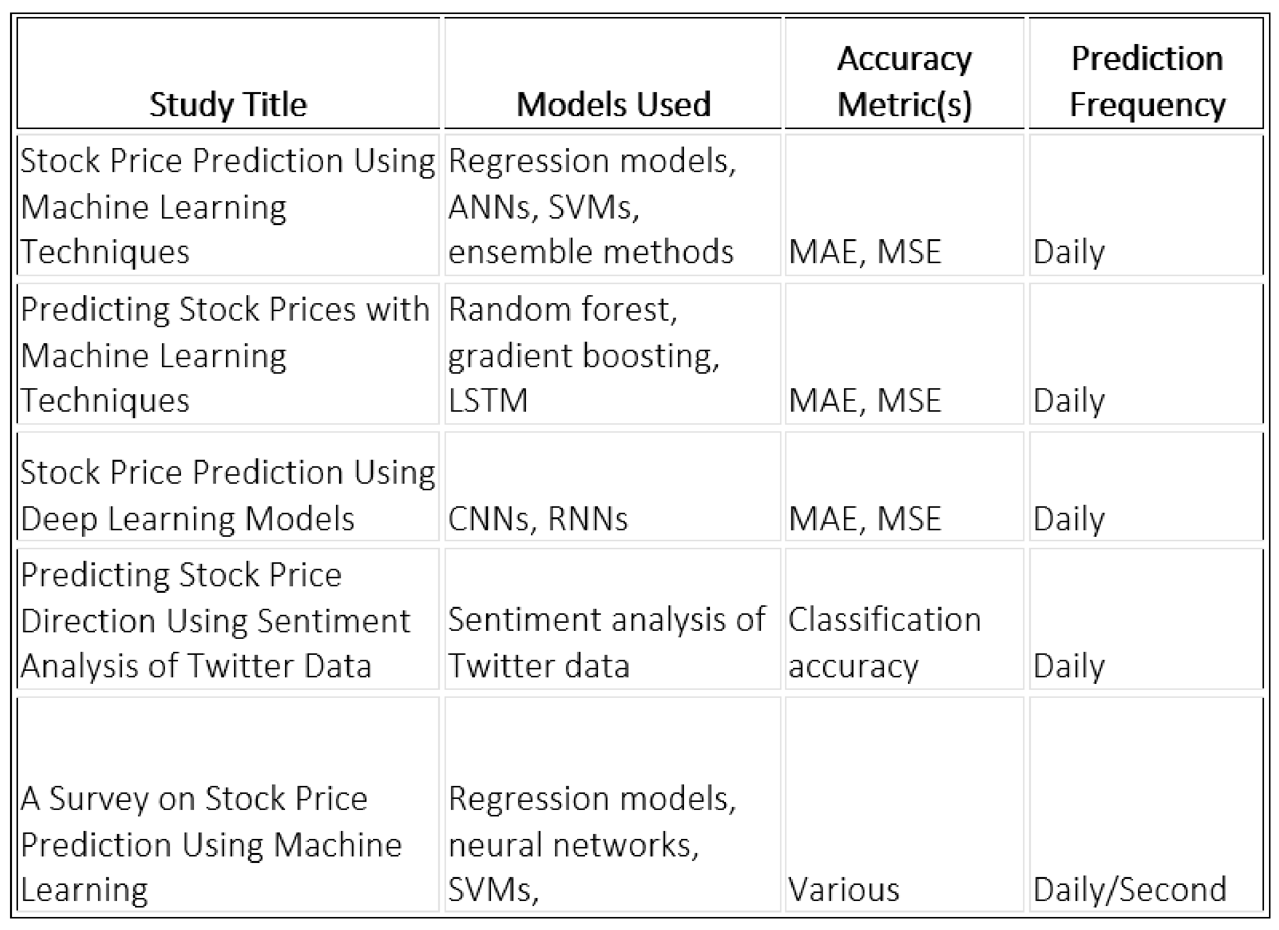
1. Removes the Investment Bias
2. Minimizes Your Losses
3. Assures Consistency
4. Gives a Better Idea about Entry and Exit Points
5. Allows the Smart Way of Making Money

TECHNICAL ANALYSIS

In this technique, machine learning models can be trained to forecast the stock movement or the direction of prices through an analysis of historical market data such as prices and volumes.

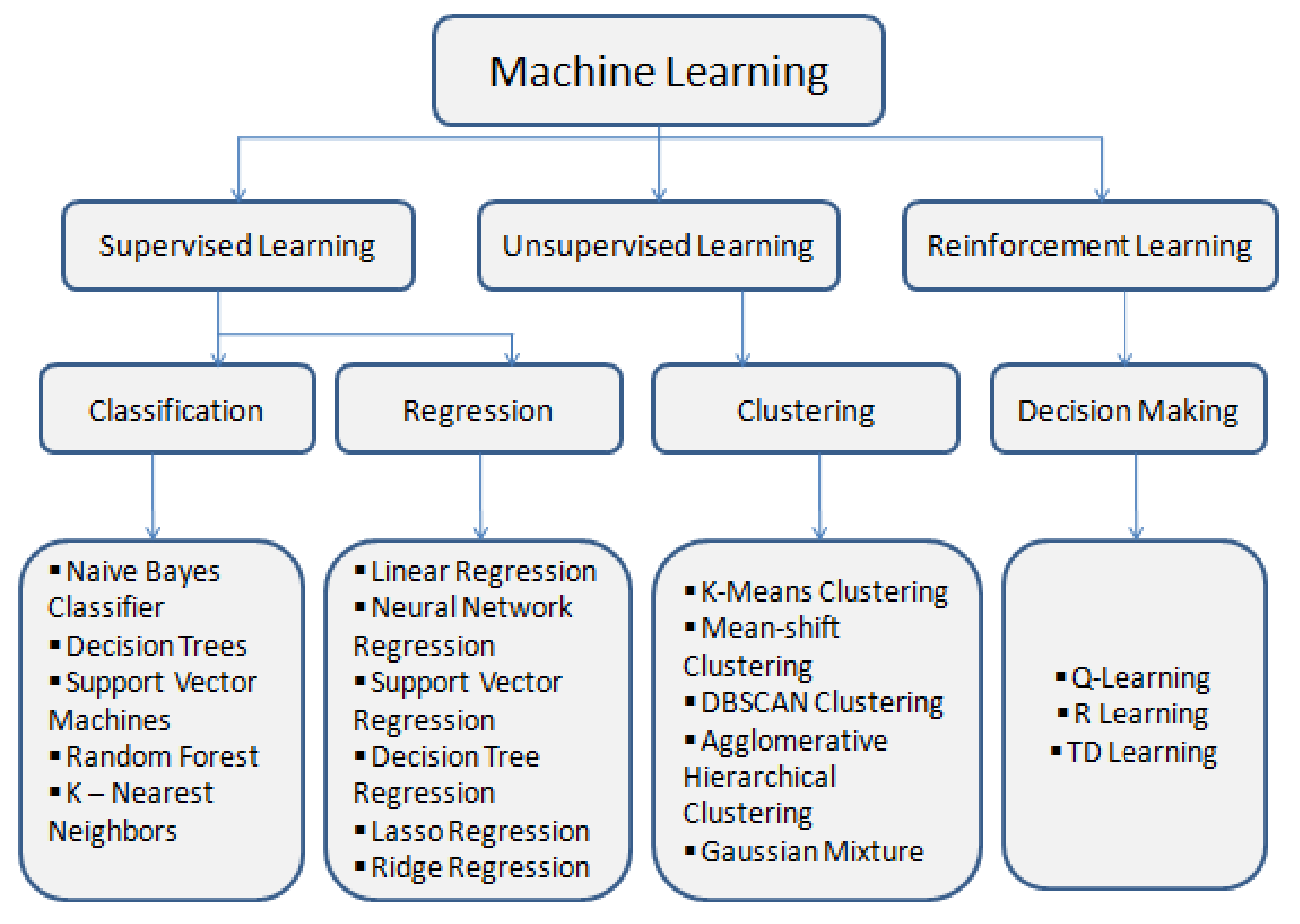


LITERATURE REVIEW



Study Title	Models Used	Accuracy Metric(s)	Prediction Frequency
Stock Price Prediction Using Machine Learning Techniques	Regression models, ANNs, SVMs, ensemble methods	MAE, MSE	Daily
Predicting Stock Prices with Machine Learning Techniques	Random forest, gradient boosting, LSTM	MAE, MSE	Daily
Stock Price Prediction Using Deep Learning Models	CNNs, RNNs	MAE, MSE	Daily
Predicting Stock Price Direction Using Sentiment Analysis of Twitter Data	Sentiment analysis of Twitter data	Classification accuracy	Daily
A Survey on Stock Price Prediction Using Machine Learning	Regression models, neural networks, SVMs,	Various	Daily/Second

MACHINE LEARNING ALGORITHMS



Machine Learning

Supervised Learning

Unsupervised Learning

Reinforcement Learning

Classification

Regression

Clustering

Decision Making

- Naive Bayes Classifier
- Decision Trees
- Support Vector Machines
- Random Forest
- K – Nearest Neighbors

- Linear Regression
- Neural Network Regression
- Support Vector Regression
- Decision Tree Regression
- Lasso Regression
- Ridge Regression

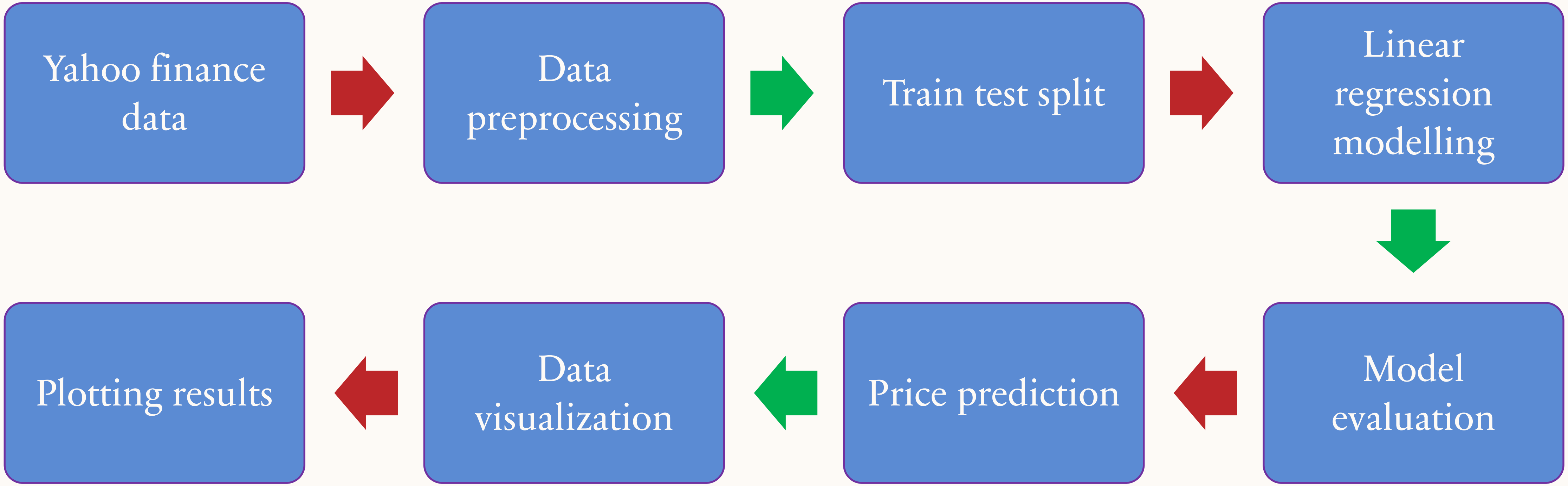
- K-Means Clustering
- Mean-shift Clustering
- DBSCAN Clustering
- Agglomerative Hierarchical Clustering
- Gaussian Mixture

- Q-Learning
- R Learning
- TD Learning

ALGORITHM

- Import libraries
- Download historical stock price data
- Extract ‘close’ price
- Create ‘next close’ column
- Drop last row with NaN value
- Feature (X) and target (Y)
- Convert to NumPy array
- Split data into training and testing sets
- Create linear regression model
- Train the model
- Predict stock prices
- Evaluate the model
- Predict future stock price
- End

DATA FLOW DIAGRAM



Yahoo finance data

Data preprocessing

Train test split

Linear regression modelling

Model evaluation

Price prediction

Data visualization

Plotting results

MODULE DESCRIPTION

STOCK PRICE DATA FROM YAHOO FINANCE

Data is obtained from Yahoo Finance using the **yfinance** library.
This step involves fetching historical stock price data for a specific stock symbol.

DATA PREPROCESSING

The collected data will be preprocessed to handle missing values, outliers, and to normalize or scale the data as required. Feature engineering will also be performed to create relevant indicators or features for the prediction model

MODEL TRAINING AND TESTING

The data will be divided into a training set and a testing set. The selected model will be trained on the training data and tested on the testing data to evaluate its performance.

MODEL SELECTION

Various machine learning algorithms such as Linear Regression.
Linear regression is a type of supervised machine learning algorithm that computes the linear relationship between a dependent variable and one or more independent features.

EVALUATION METRICS

Evaluation Metrics: The performance of the prediction model will be assessed using metrics like Mean Squared Error (MSE), and R-squared (R2) to measure accuracy and model goodness of fit.

LINEAR REGRESSION ANALYSIS

Linear regression is a type of supervised machine learning algorithm that computes the linear relationship between a dependent variable and one or more independent features.

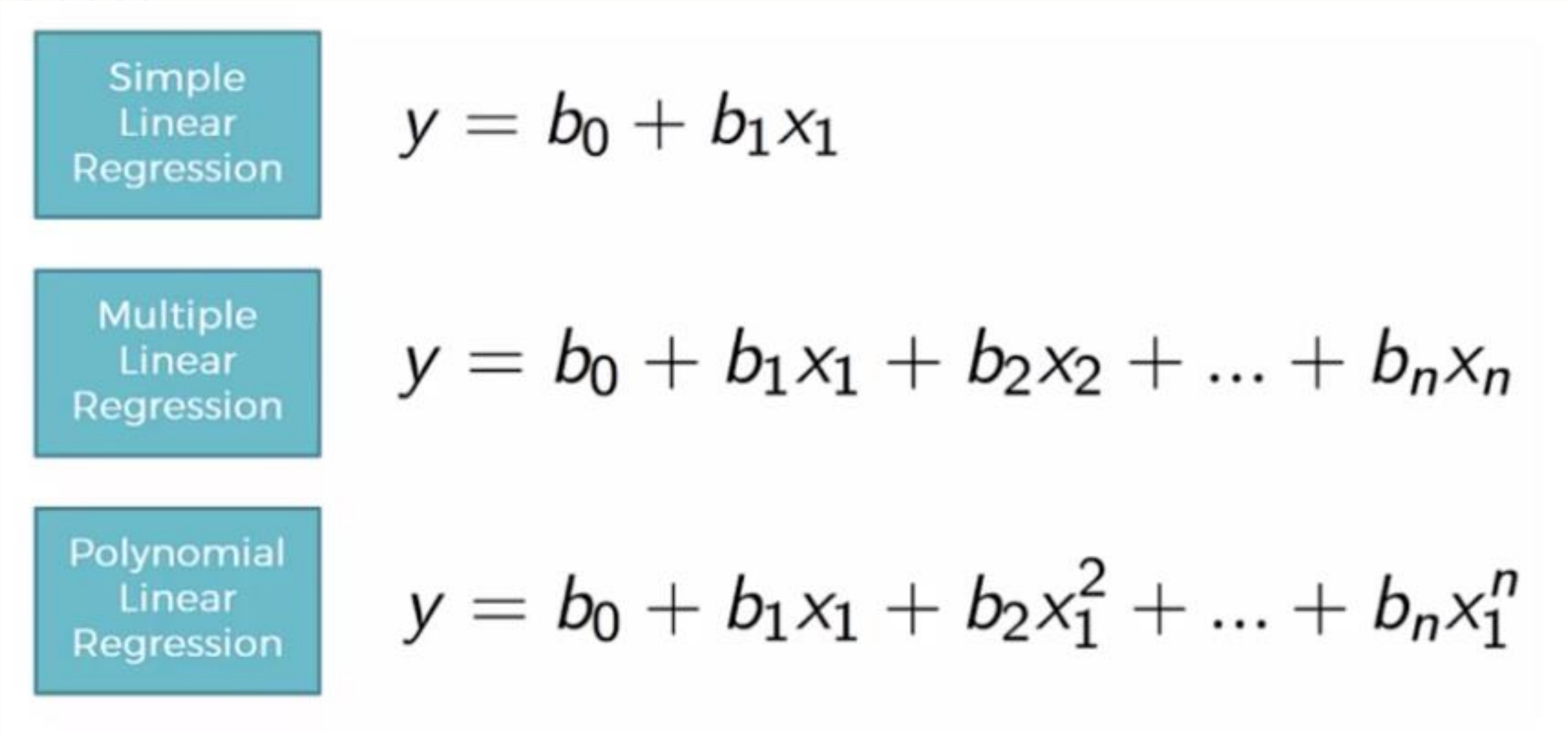
When the number of the independent feature, is 1 then it is known as Univariate Linear regression, and in the case of more than one feature, it is known as multivariate linear regression.

The goal of the algorithm is to find the best linear equation that can predict the value of the dependent variable based on the independent variables.

The equation provides a straight line that represents the relationship between the dependent and independent variables.

The slope of the line indicates how much the dependent variable changes for a unit change in the independent variable(s).

LINEAR REGRESSION



Simple Linear Regression

$$y = b_0 + b_1 x_1$$

Multiple Linear Regression

$$y = b_0 + b_1 x_1 + b_2 x_2 + \ldots + b_n x_n$$

Polynomial Linear Regression

$$y = b_0 + b_1 x_1 + b_2 x_1^2 + \ldots + b_n x_1^n$$

SVM MODEL

Support Vector Regression (SVR) is a supervised learning algorithm used for regression tasks, particularly in cases where linear regression models may not be effective due to non-linear relationships between the input features and the target variable. SVR extends the principles of Support Vector Machines (SVM) to the regression context.

Objective:

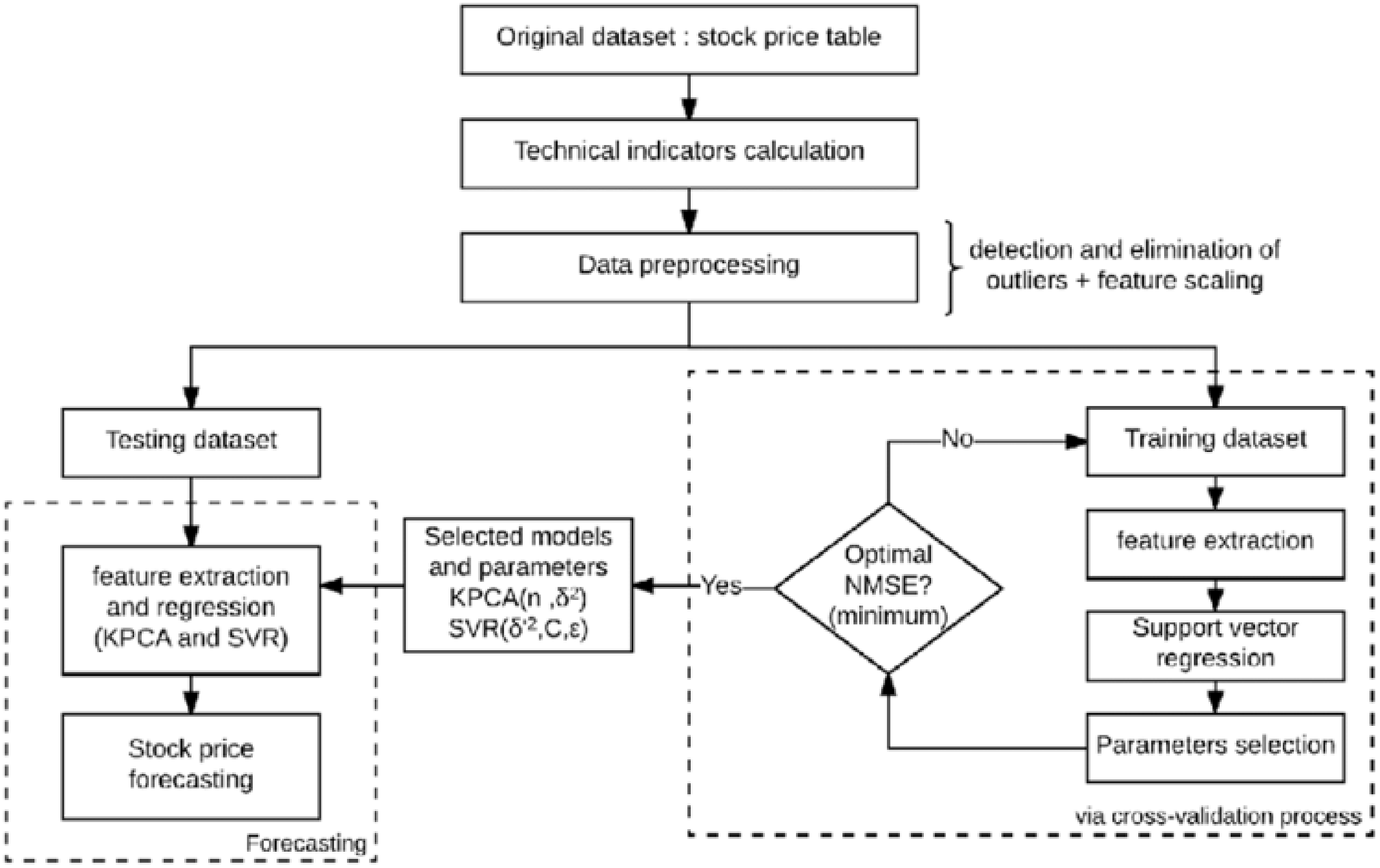
- The objective of SVR is to find a function that approximates the mapping from input features to the continuous target variable with minimal error.
- Unlike traditional regression models that aim to minimize prediction errors directly, SVR focuses on minimizing the margin of error around the predicted values, ensuring that most data points fall within a specified margin of tolerance.

Kernel Trick:

- SVR employs a kernel trick to implicitly map input features into a higher-dimensional space where the relationship between features and the target variable may be linear.
- Common kernel functions used in SVR include the Radial Basis Function (RBF), polynomial, and sigmoid kernels.
- The choice of kernel function and associated parameters (e.g., kernel width for RBF kernel) significantly influences the flexibility and performance of the SVR model.

- **Margin of Tolerance**:
  - In SVR, the margin of tolerance around the predicted values is controlled by two parameters:
    - Epsilon (ε): Specifies the width of the margin, indicating the acceptable deviation of predicted values from the true target values.
    - C: Regularization parameter that balances the trade-off between maximizing the margin and minimizing the error on the training data. Higher values of C prioritize minimizing training error, potentially leading to overfitting.
- **Loss Function**:
  - SVR minimizes a loss function that penalizes deviations of predicted values from the true target values while ensuring that deviations within the margin of tolerance (ε) are ignored.
  - The loss function typically consists of two components:
    - Regression loss: Measures the error between predicted and true target values.
    - Regularization term: Penalizes large coefficients or deviations from the margin of tolerance, promoting a simpler model with a wider margin.
- **Support Vectors**:
  - Support vectors are the data points that lie on or within the margin of tolerance and contribute to defining the decision boundary of the SVR model.
  - These are the critical data points that influence the model's predictions and are instrumental in capturing the underlying patterns in the data.

Model Working


Original dataset : stock price table
Technical indicators calculation
Data preprocessing
detection and elimination of outliers + feature scaling
Testing dataset
Training dataset
No
feature extraction
Optimal NMSE? (minimum)
Yes
Support vector regression
Parameters selection
via cross-validation process
Selected models and parameters KPCA(n ,δ²) SVR(δ'², C, ε)
feature extraction and regression (KPCA and SVR)
Stock price forecasting
Forecasting

STATISTICAL MEASURES

MEAN SQUARE ERROR

- The Mean Squared Error measures how close a regression line is to a set of data points. It is a risk function corresponding to the expected value of the squared error loss.
- Mean square error is calculated by taking the average, specifically the mean, of errors squared from data as it relates to a function
- A larger MSE indicates that the data points are dispersed widely around its central moment (mean)
- A smaller MSE is preferred because it indicates that your data points are dispersed closely around its central moment (mean).

$$MSE = \frac{1}{n}\Sigma\underbrace{\left(y-\widehat{y}\right)^2}_{\text{The square of the difference between actual and predicted}}$$

Lesser the MSE => Smaller is the error => Better the estimator.

STATISTICAL MEASURES

R^2 OR COEFFICIENT OF DETERMINATION

R-squared evaluates the scatter of the data points around the fitted regression line. For the same data set, higher R-squared values represent smaller differences between the observed data and the fitted values.

R-squared is always between 0 and 100%:

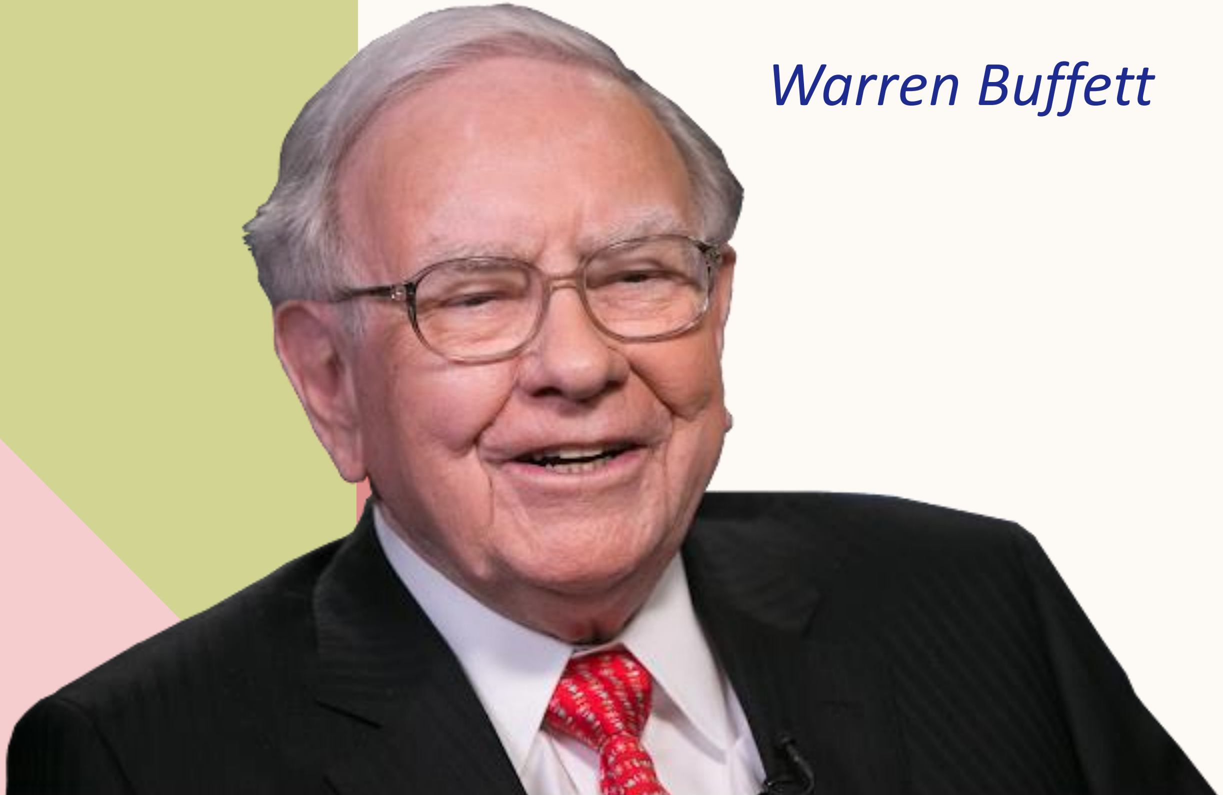
- 0% represents a model that does not explain any of the variation in the response variable around its mean. The mean of the dependent variable predicts the dependent variable as well as the regression model.
- 100% represents a model that explains all the variation in the response variable around its mean.

$$R^2 = 1 - \frac{\overset{\text{Sum Squared Regression Error}}{SS_{Regression}}}{\underset{\text{Sum Squared Total Error}}{SS_{Total}}}$$

The larger the R^2, the better the regression model fits your observations.

> “A MARKET DOWNTURN DOESN’T BOTHER US. IT IS AN OPPORTUNITY TO INCREASE OUR OWNERSHIP OF GREAT COMPANIES WITH GREAT MANAGEMENT AT GOOD PRICES.”

Warren Buffett



PYTHON LIBRARIES

- Pandas
- Numpy
- Scikit learn
- Matplotlib
- Yfinance
- Flask
- Tweety

Pandas

- **Description:**
  - Pandas is an open-source data manipulation and analysis library for Python.
  - It provides data structures such as Series and DataFrame for efficient data handling and analysis.
- **Key Features:**
  - **DataFrame:** A 2-dimensional labeled data structure with columns that can be of different types.
  - **Data Cleaning:** Provides functions for handling missing data, filtering, and transforming data.
  - **Data Alignment:** Supports automatic and explicit data alignment.
  - **Data Input/Output:** Can read data from various file formats like CSV, Excel, SQL databases, and more.

NumPy

- **Description:**
  - NumPy is a fundamental package for scientific computing in Python.
  - It provides support for large, multi-dimensional arrays and matrices, along with mathematical functions to operate on these arrays.
- **Key Features:**
  - **Arrays:** Efficient and fast array object for numerical operations.
  - **Broadcasting:** Allows operations on arrays of different shapes and sizes.
  - **Linear Algebra:** Provides functions for linear algebra operations.
  - **Random:** Includes tools for random number generation.

Scikit-learn (sklearn):

- **Description:**
  - Scikit-learn is a machine learning library in Python that provides simple and efficient tools for data analysis and modeling.
  - It is built on NumPy, SciPy, and Matplotlib.
- **Key Features:**
  - **Supervised Learning:** Includes algorithms for classification, regression, and ensemble methods.
  - **Unsupervised Learning:** Provides clustering, dimensionality reduction, and density estimation algorithms.
  - **Model Selection:** Tools for model selection, hyperparameter tuning, and evaluation.
  - **Data Preprocessing:** Utilities for feature extraction, scaling, and transformation.

Matplotlib:

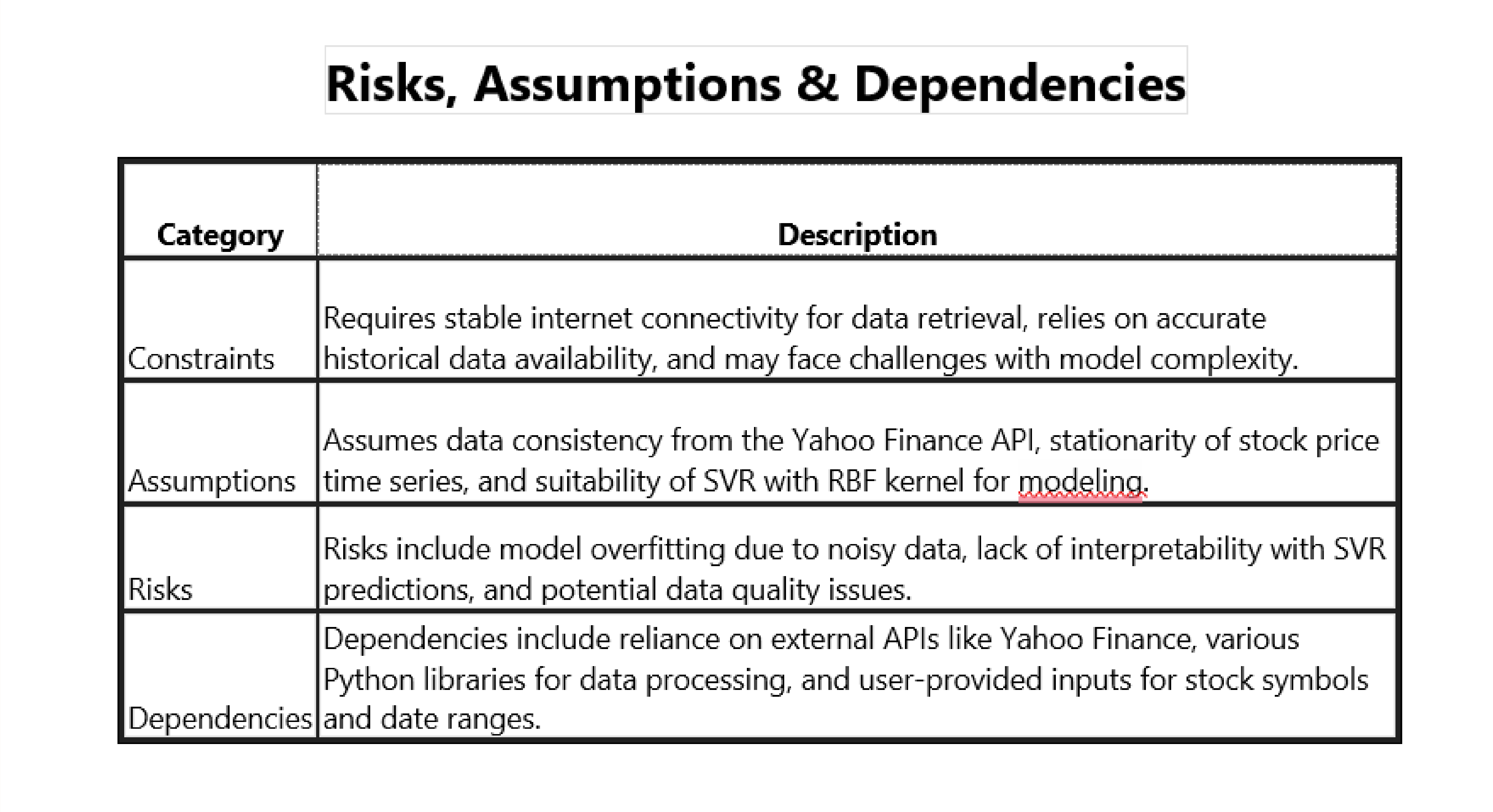
- **Description:**
  - Matplotlib is a 2D plotting library for Python that produces static, animated, and interactive visualizations.
- **Key Features:**
  - **Plots and Charts:** Provides a variety of plot types, including line plots, scatter plots, bar plots, and more.
  - **Customization:** Offers extensive customization options for labels, colors, and styles.
  - **Subplots:** Supports creating multiple plots in a single figure.
  - **Exporting:** Can save figures in various formats (e.g., PNG, PDF).

Yfinance :

- **Description:**
  - **Yfinance** is a python package that enables us to fetch historical market data from Yahoo Finance API in a Pythonic way
- **Key Features:**
  - **Historical Data:** Allows fetching historical market data for stocks, indices, and more.
  - **Real-time Data:** Provides real-time stock quotes.
  - **Data Adjustment:** Supports dividend and split adjustments.

Flask:

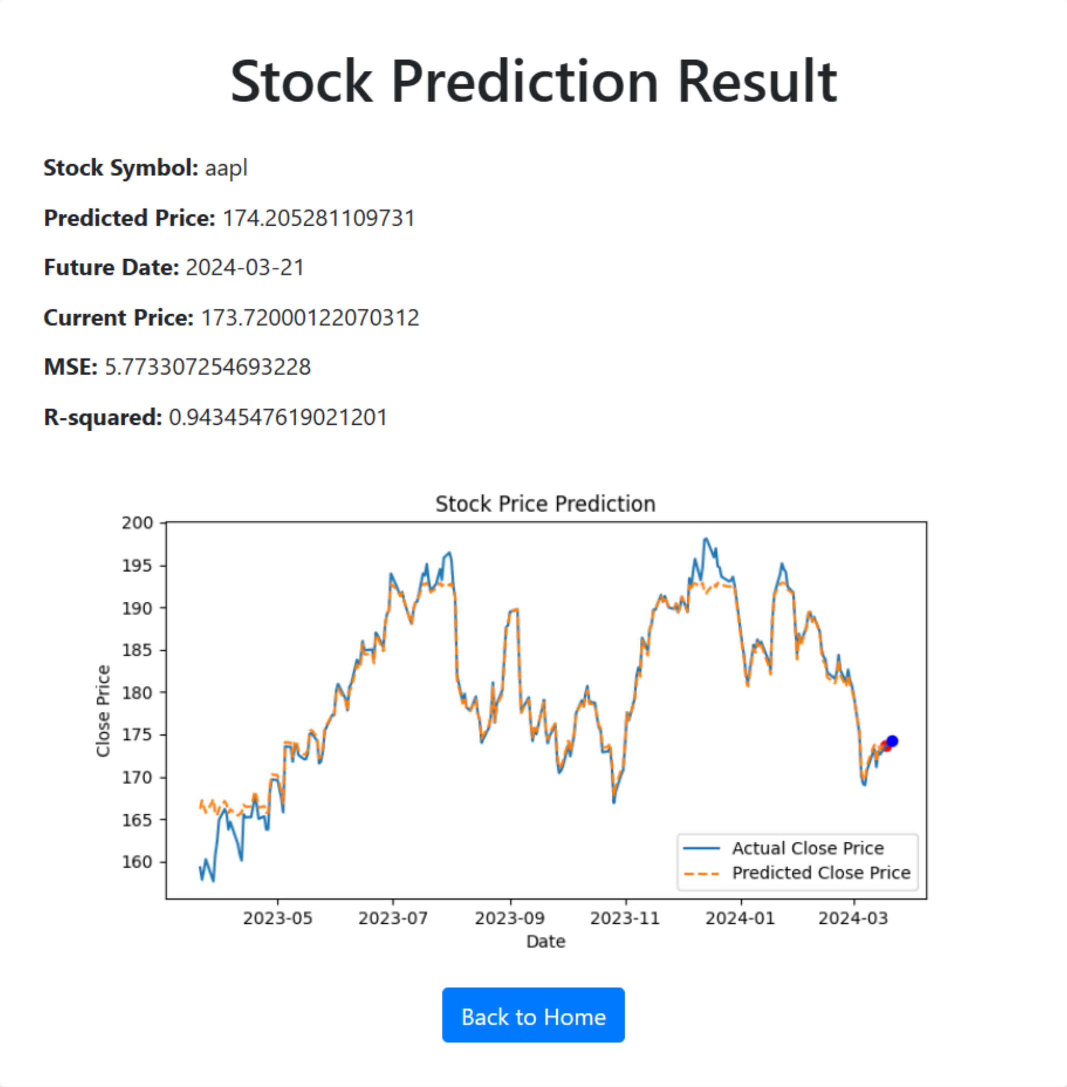
- **Description:**
  - Flask is a micro web framework, meaning it provides only the essential tools needed to get a web application up and running.
  - It is based on the Werkzeug WSGI toolkit and the Jinja2 template engine.
- **Key Features:**
  - **Routing:** Easily define URL patterns and associate them with functions (views) that handle the incoming requests.
  - **Templates:** Use Jinja2 templates for HTML rendering, allowing dynamic content generation.
  - **HTTP Methods:** Support for handling different HTTP methods (GET, POST, etc.) for routing.
  - **Extensions:** Flask has a modular design and allows the use of extensions for added functionality.



Risks, Assumptions & Dependencies

Category	Description
Constraints	Requires stable internet connectivity for data retrieval, relies on accurate historical data availability, and may face challenges with model complexity.
Assumptions	Assumes data consistency from the Yahoo Finance API, stationarity of stock price time series, and suitability of SVR with RBF kernel for modeling.
Risks	Risks include model overfitting due to noisy data, lack of interpretability with SVR predictions, and potential data quality issues.
Dependencies	Dependencies include reliance on external APIs like Yahoo Finance, various Python libraries for data processing, and user-provided inputs for stock symbols and date ranges.

PREDICTION FOR APPLE



Stock Prediction Result

Stock Symbol: aapl

Predicted Price: 174.205281109731

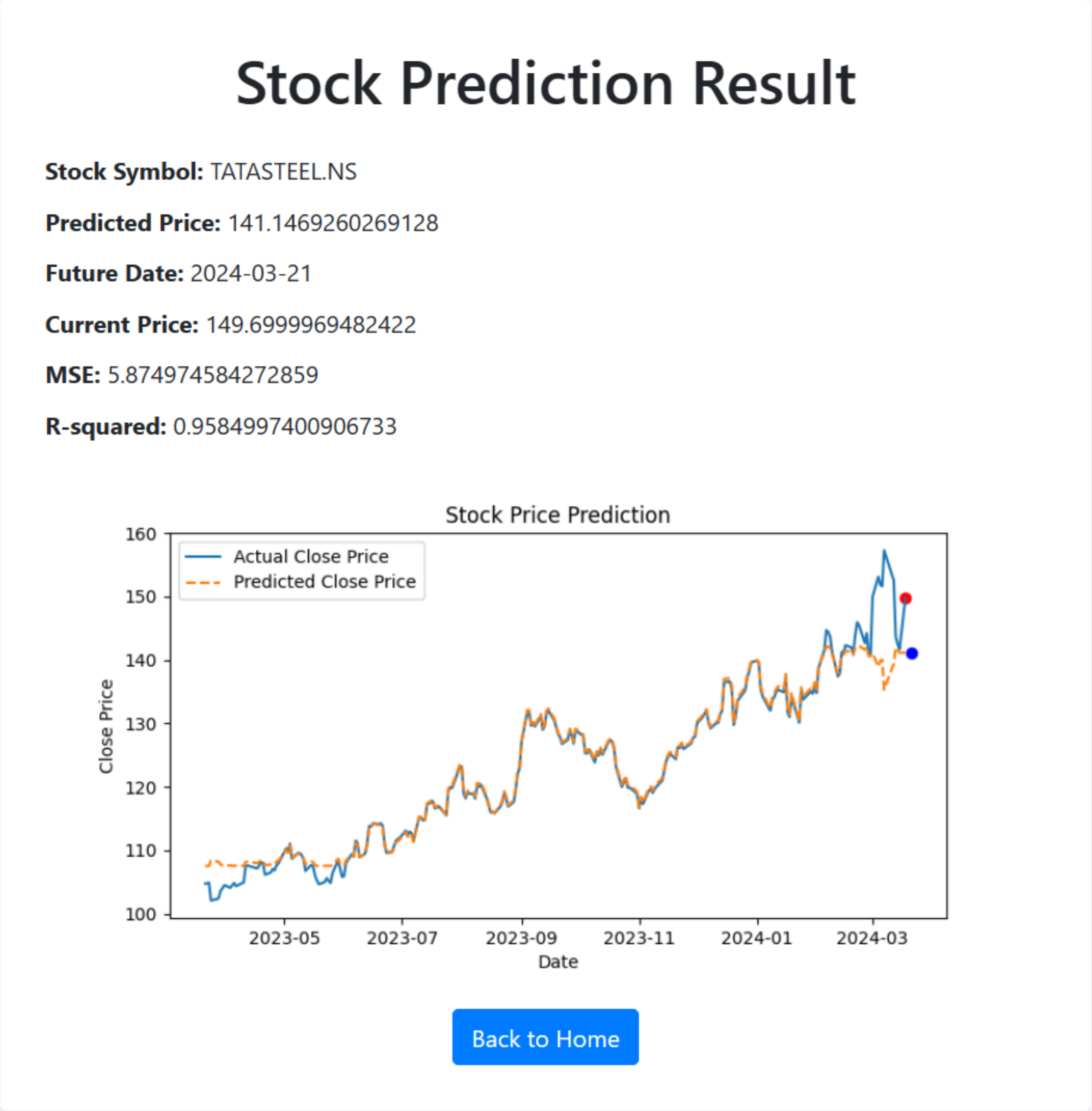
Future Date: 2024-03-21

Current Price: 173.72000122070312

MSE: 5.773307254693228

R-squared: 0.9434547619021201

PREDICTION FOR TATA STEEL



Stock Prediction Result

Stock Symbol: TATASTEEL.NS

Predicted Price: 141.1469260269128

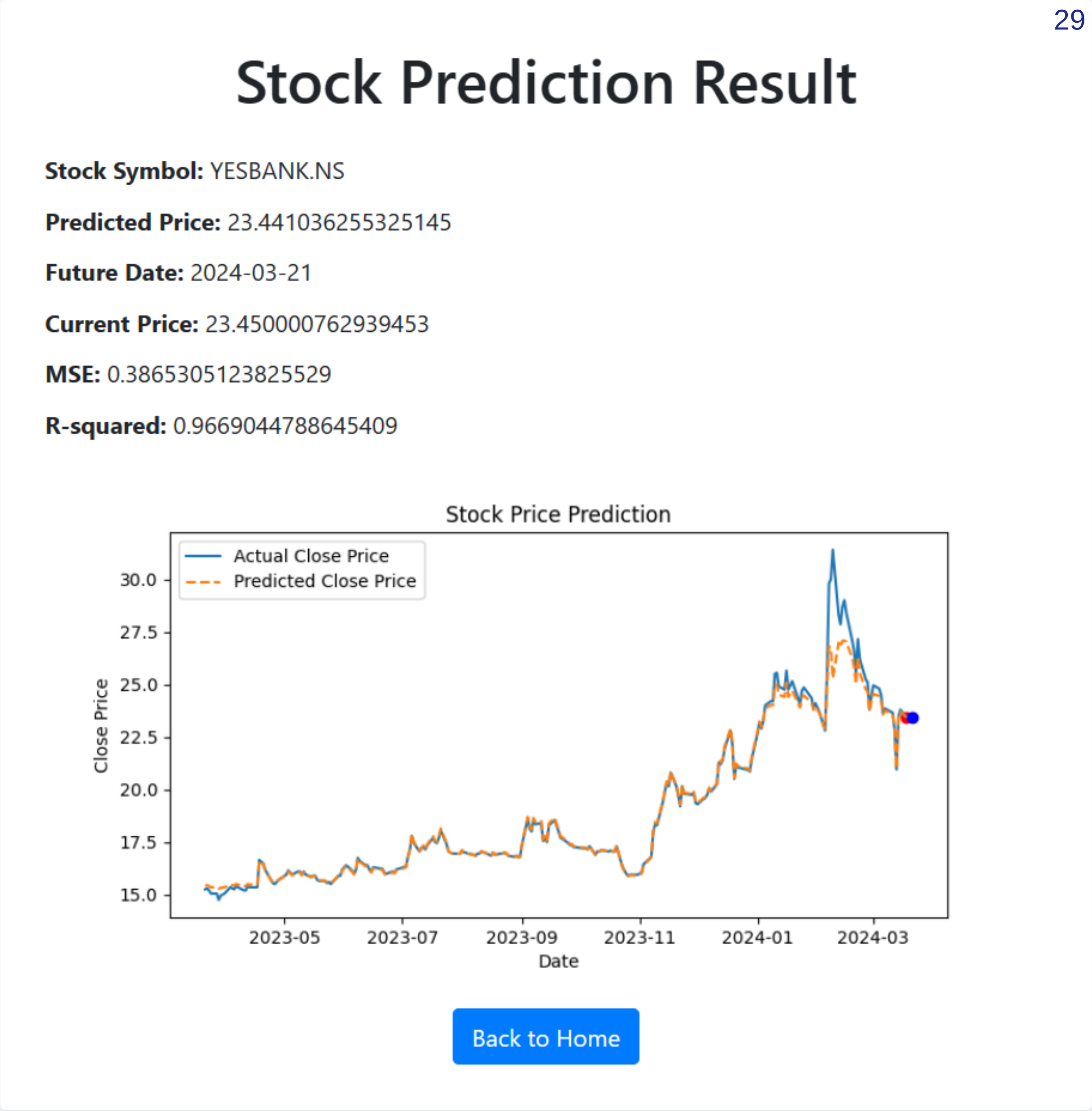
Future Date: 2024-03-21

Current Price: 149.6999969482422

MSE: 5.874974584272859

R-squared: 0.9584997400906733

PREDICTION FOR YES BANK



Stock Prediction Result

Stock Symbol: YESBANK.NS

Predicted Price: 23.441036255325145

Future Date: 2024-03-21

Current Price: 23.450000762939453

MSE: 0.3865305123825529

R-squared: 0.9669044788645409

WHAT COULD POSSIBLY GO WRONG?

1.Limited Model Complexity:

The linear relationship between the closing prices may not capture more complex patterns or nonlinear relationships in the data. This simplicity might lead to underfitting, where the model fails.

2. Overfitting to Historical Data:

While the model is trained on historical data, it might overfit to specific patterns in that data, resulting in poor generalization to new, unseen data. Overfitting occurs when the model learns noise in the training data rather than the underlying patterns.

3. Market Noise and Randomness:

The stock market contains inherent noise and randomness, making it challenging to discern meaningful patterns from mere fluctuations.

4. External Factors:

External events, such as unexpected news or global economic changes, can have a significant impact on stock prices.

FUTURE ENHANCEMENT

FEATURE ENGINEERING

- **Additional Indicators:** Incorporate other financial indicators (e.g., moving averages, technical indicators) as features to capture more complex relationships in the data.
- **Lagged Features:** Include lagged versions of other relevant features to account for potential time dependencies.

ADVANCED MODELS

- **Time Series Models:** Explore time series models like ARIMA or SARIMA for better handling of temporal patterns in stock prices.
- **Machine Learning Ensemble Methods:** Experiment with ensemble methods like Random Forest or Gradient Boosting for improved predictive performance.

HYPERPARAMETER TUNING

Fine-tune hyperparameters of the linear regression model or other chosen models to optimize their performance on the given dataset.

FUTURE ENHANCEMENT

CROSS-VALIDATION

Implement cross-validation techniques to obtain a more robust estimate of the model's performance and reduce overfitting.

HANDLING OUTLIERS AND ANOMALIES

Develop strategies to handle outliers and anomalies in the data, which can significantly impact the model's performance.

NEWS AND SENTIMENT ANALYSIS

Integrate natural language processing (NLP) techniques to analyze financial news and sentiment data, as these can impact stock prices.

SUMMARY

In this project, we leveraged the power of machine learning to predict stock prices based on historical data obtained from Yahoo Finance. By implementing a linear regression model. Through careful data preprocessing, train-test splitting, and model evaluation, the project achieved a predictive framework capable of forecasting future stock prices.

In conclusion, predicting stock prices is no walk in the park. Our project is a stepping stone, a humble attempt to unravel the complexities of financial markets. As we move forward, we remain cognizant of the challenges, embrace the experimental nature of our approach, and eagerly explore avenues for improvement.

THANK YOU

Submitted by

Name: Harshita

Course: MCA

Year:1st Semester 1st Year