# Stock Prediction Using Regression

This project predicts stock closing prices of Tata Global using regression models. We leverage historical NSE data and advanced techniques for precision forecasting.

**by Orbin BCR70**

# Project Overview & Dataset Insights

## Dataset

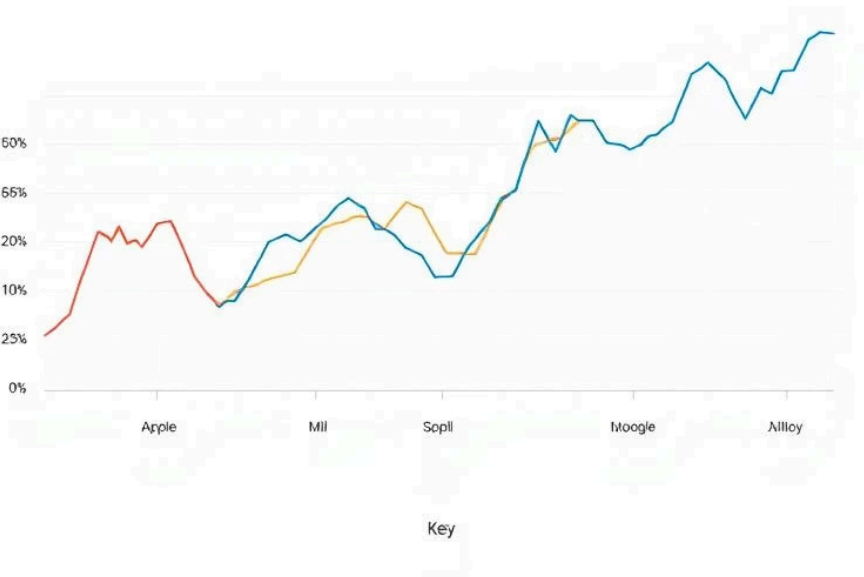
NSE Tata Global stock data with key price and volume metrics.

## Key Variables

- Date, Open, High, Low, Close prices
- Last traded price, Trade quantity, Turnover

## Goal

Predict closing price accurately using historical features.




# Exploratory Data Analysis (EDA)

### Price Trends

Line plots display closing price patterns over time.

### Volume Distribution

Histograms reveal trade quantity frequency and spikes.

### Outlier Detection

Boxplots highlight unusual price observations for Open, High, Low, Close.

# Feature Engineering & Scaling

## Selected Features

- Open, High, Low prices
- Total Trade Quantity
- Turnover (Lacs)

## Scaling Method

StandardScaler applied to normalize feature ranges for model efficiency.

Ensures uniformity and improves learning stability.


# Modeling: Linear Regression & Cross-Validation

1

## Linear Regression

Trained on 80% data, tested on 20% to predict closing price.

2

## 5-Fold Cross-Validation

Validated model stability using $R^2$ scoring metric across folds.

3

## Evaluation Metrics

- $R^2$ Score, MSE, MAE
- Assessed prediction accuracy and errors

# Advanced Modeling: Random Forest & Hyperparameter Tuning

## Grid Search

Tuned n_estimators and max_depth with exhaustive 5-fold cv search.

## Randomized Search

Explored broader hyperparameter ranges efficiently with 10 iterations.

Achieved best model performance.

# Feature Importance & Model Interpretation

## Top Features

Open, High, Low prices contribute most to prediction accuracy.

## Lower Impact Features

Turnover and Trade Quantity have less influence and can be considered for removal.

## Implications

Simplifying model by focusing on key price inputs may improve efficiency.


# Summary & How to Run

## 1 Summary

Random Forest with Randomized Search yielded highest accuracy ($R^2$ ~ 0.99983).

## 2 Setup Steps

1. Install required Python libraries.
2. Download dataset and place alongside notebook.
3. Run regression.ipynb in Jupyter.