NIFTY 50 Stock Price Prediction & Pairs Trading Classification

Aditya Moza 2022035   Manit Kaushik 2022277

1 Introduction

The objective of our project is to do an analysis of the stock prices of companies listed in the NIFTY 50 index using machine learning techniques. The NIFTY 50 index comprises India's top 50 companies across various sectors, making it a significant benchmark for the Indian stock market. This project is divided into three main sections: stock price prediction, stock classification based on growth, and classification of profitable pairs for pairs trading. The datasets used for our project is NIFTY-50 Stock Market Data (2000 - 2021) and the Yahoo Finance Python API.

2 Background

Analyzing stock prices in the NIFTY 50 index holds significant relevance for investors, analysts, and policymakers alike. Understanding the dynamics of these leading companies provides insights into broader market trends, economic conditions, and investment opportunities within India's stock market landscape.

Numerous studies have explored stock price prediction and classification using machine learning methods, particularly in the context of financial markets. Research often encompasses various algorithms, including regression models, neural networks, and ensemble methods, to forecast stock prices and classify stocks based on growth potential. Additionally, the concept of pairs trading, aimed at exploiting market inefficiencies, has been extensively studied, with researchers investigating diverse strategies and methodologies to identify profitable pairs.

3 Stock Price Prediction

For this task, we have implemented and compared between three well-known Machine Learning techniques, Convolutional Neural Network Long Short-Term Memory Model (**CNN-LSTM**), K-Nearest Neighbours (**KNN**) & Multi-Linear Regression (**MLR**). We have presented our findings for this section using TCS Stock.

|	Date	Symbol	Series	Prev Close	Open	High	Low	Last	Close	VWAP	Volume	Turnover	Trades	Deliverable Volume	%Deliverble
0	2004-08-25	TCS	EQ	850.00	1198.70	1198.70	979.00	985.00	987.95	1008.32	17116372	1.725876e+15	NaN	5206360	0.3042
1	2004-08-26	TCS	EQ	987.95	992.00	997.00	975.30	976.85	979.00	985.65	5055400	4.982865e+14	NaN	1294899	0.2561
2	2004-08-27	TCS	EQ	979.00	982.40	982.40	958.55	961.20	962.65	969.94	3830750	3.715586e+14	NaN	976527	0.2549
3	2004-08-30	TCS	EQ	962.65	969.90	990.00	965.00	986.40	986.75	982.65	3058151	3.005106e+14	NaN	701664	0.2294
4	2004-08-31	TCS	EQ	986.75	986.50	990.00	976.00	987.80	988.10	982.18	2649332	2.602133e+14	NaN	695234	0.2624
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
4134	2021-04-26	TCS	EQ	3109.50	3102.05	3153.00	3088.05	3100.05	3100.80	3122.35	2490260	7.775465e+14	87012.0	1345353	0.5402
4135	2021-04-27	TCS	EQ	3100.80	3106.00	3136.10	3103.00	3132.75	3132.75	3122.18	1471417	4.594036e+14	97077.0	849767	0.5775
4136	2021-04-28	TCS	EQ	3132.00	3149.95	3151.00	3112.00	3123.00	3124.10	3125.72	1639037	5.123176e+14	76017.0	847762	0.5172
4137	2021-04-29	TCS	EQ	3124.10	3145.60	3157.35	3105.50	3114.45	3115.25	3130.83	1621395	5.076306e+14	98275.0	837567	0.5166
4138	2021-04-30	TCS	EQ	3115.25	3099.00	3132.05	3020.00	3040.00	3035.65	3063.19	3072305	9.411057e+14	157829.0	1942473	0.6323

4139 rows × 15 columns

3.1 CNN-LSTM

This model employs a short history of a stock, utilizing the prices of the previous 5 days as features to forecast future prices. It adopts a hybrid architecture, integrating Convolutional Neural Networks (CNNs) with an LSTM layer, followed by a linear layer for prediction. The model is trained using mean squared error loss and the Adam optimizer over 20 epochs.

Table 1: LSTM Data Table Used

| Date	Close	Close t-1	Close t-2	Close t-3	Close t-4	Close t-5
2004-09-01	987.90	988.10	986.75	962.65	979.00	987.95
2004-09-02	993.65	987.90	988.10	986.75	962.65	979.00
2004-09-03	997.85	993.65	987.90	988.10	986.75	962.65
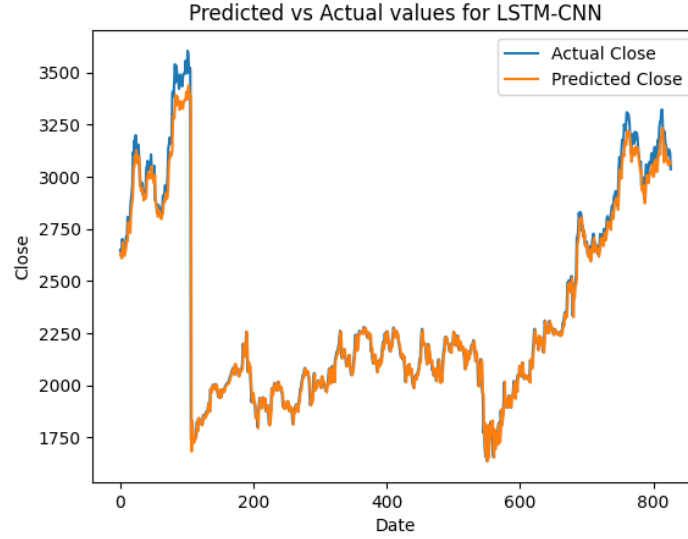
3.2 Multi-Linear Regression

In this method, a multi-linear regression (MLR) model is applied to predict the Close Price of a stock using various features including "Open," "High," "Low," "Volume," and "Turnover" from the dataset.
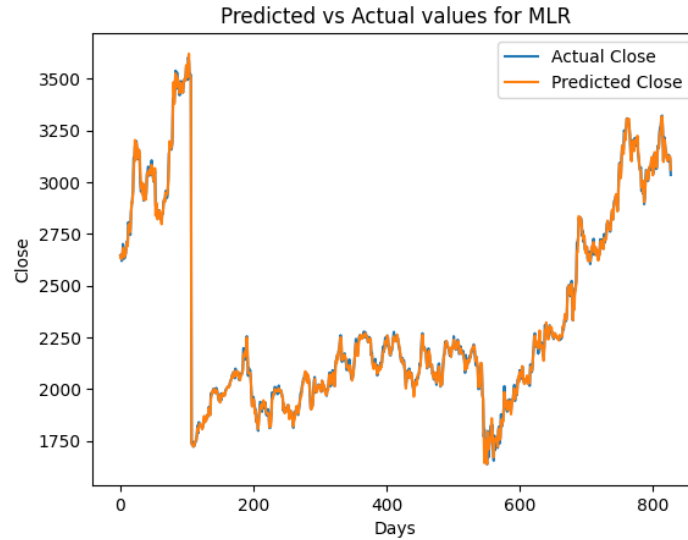
3.3 K-Nearest Neighbours

In this method, the K-Nearest Neighbors Regression (KNN) algorithm is applied for predicting the stock's closing price. The KNN algorithm works by finding the 'k' nearest data points in the feature space and averaging their target values to make predictions.
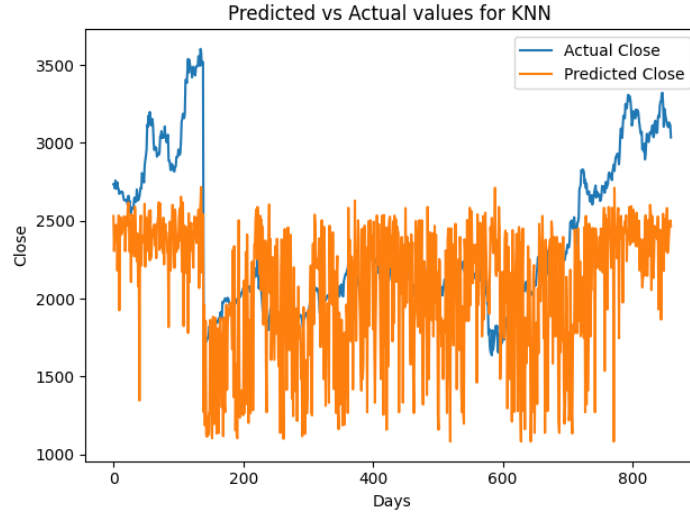
| Method	MSE on Test Dataset
CNN-LSTM	**0.04678102**
MLR	**0.00010230**
KNN	**0.41147872**

From the Predicted v/s Actual Values Graphs and the MSE Errors Table, we can conclude that for this dataset our MLR Model performs the best out of the three models created and KNN performs the worst.

4 Stock Classification Based on Growth

In this section, we have used K Nearest Neighbours and Decision Trees to classify whether the price of a stock will go higher than the price of previous of day or lower.

4.1 K-Nearest Neighbours

In this method, the K-Nearest Neighbors (KNN) algorithm is used for a classification problem. The objective is to predict whether to Buy (+1) or Sell (-1), essentially predicting whether the stock will rise or fall. The results are as follows:

Confusion Matrix for KNN Classifier
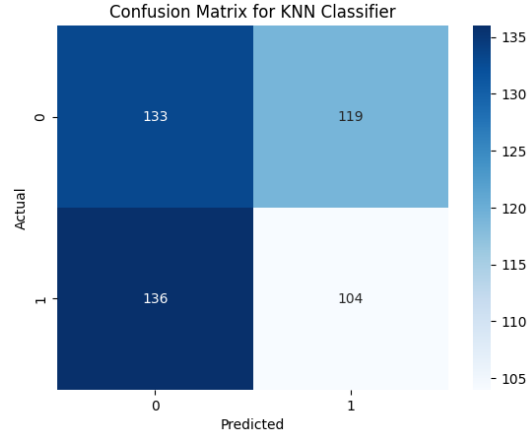
4.2 Decision Trees

In this method, a Decision Trees Classifier is employed to predict stock price movements based on features such as "Open - Close," "High - Low," "Volume," and "Turnover." The target variable is determined by comparing the closing price of the current day with that of the next day. The labels of +1 and -1 mean the same here as it did in the KNN Model. Since this model gives a different result every time we took the average of training and testing accuracies over 100 epochs. The results are as follows:


Confusion Matrix for Decision Tree Classifier
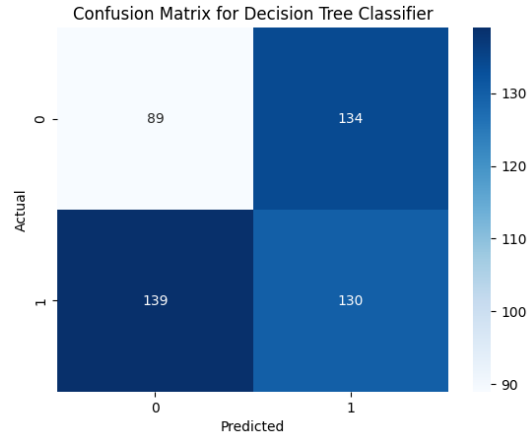
| Method	MSE on Test Dataset
KNN	**0.4817073170731707**
Dec. Trees	**0.4539024390243902**

From the Confusion Matrices and the MSE Errors Table, we can see that the accuracies for both the models are very similar and pretty low.

5 Pairs Trading Classification

Pairs trading is a popular trading strategy that involves the simultaneous buying and selling of two correlated financial instruments to profit from temporary divergences in their prices. We used the method of K Means Clustering over the volatility and returns of stocks to identify potential trading pairs. First to figure out the optimal value of K, we used the Elbow Method and located the Knee in the Distortion v/s K graph,which came to be at K=5.
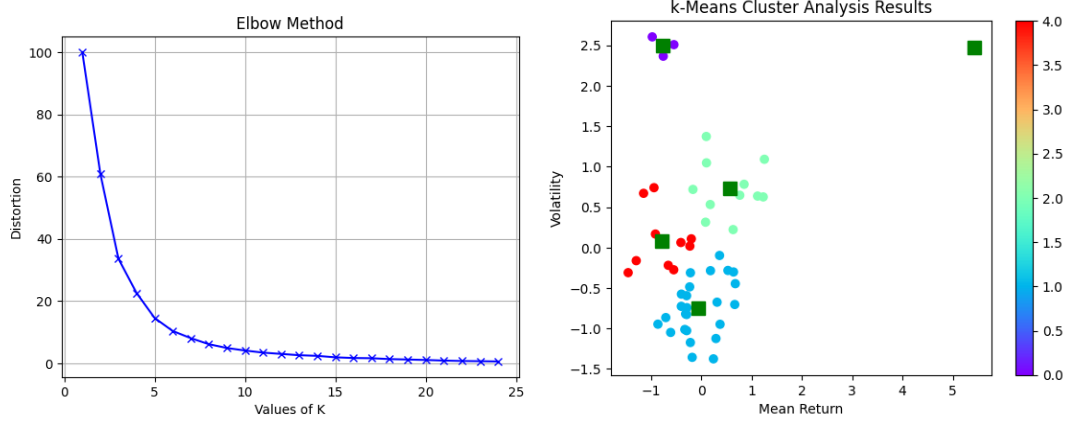
Then in each of the formed clusters, we looked at all stock pairs and ran a Cointegration Test on their stock values from 2018-22, that is if the p-value of a pair of stock was less than 0.5, we can say that they might be cointegrated. The number for these potential valid trading pairs came out to be 43. So now, to test if our potential valid trading pairs are actually profitable or not, we took each pair and their stock prices in 2023-24 and simulated the results as if they were a trading pair and calculated their Sharpe Ratio for this time. Sharpe Ratio, if comes out be positive, indicates positive returns for us and conversely if it is negative. Out of the 43 potential trading pairs, 23 pairs gave positive returns.

Profit Pairs Graph
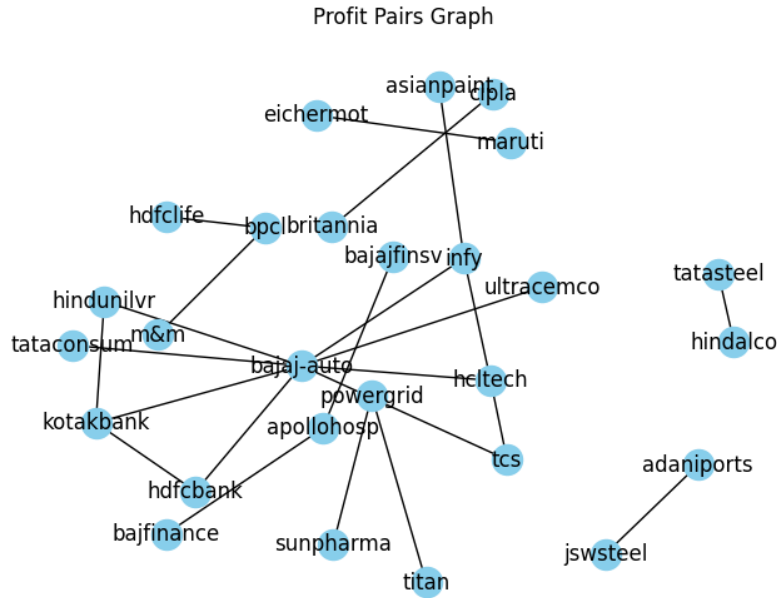
6 References

- scikit-learn.org

- pytorch.org

- algotrading101.com

- kaggle.com

- machinelearningmastery.com