

Stock Market Prediction Using Machine Learning

Notes

[Slide 1] MAIN

Hello everyone, today we're going to present our project and its details. Our project is about **Stock Market Prediction using Machine Learning** and we're doing this project under the guidance of our HOD, Dr. John Peter.

[Slide 2] Agenda

This is today's agenda that we'll be discussing today.

[Slide 3] Abstract

Stock Market is one of the many ways in which companies, businessmen and even common citizens earn money by investing in the share market. The share market is extremely volatile and changes due to factors such as political events and general economic conditions. It therefore becomes necessary to be able to predict the prices so that investors can have maximum profit in return.

[Slide 4] Introduction

To predict stock prices, we used statistical methods such as Time Series Forecasting and other systematic analysis that involved linear models. Due to the growing emergence of Artificial Intelligence, and due to its various applications, Machine Learning and its methods can be applied in order to predict the stock prices.

Before we move on to see how this can be achieved, let's briefly understand what Machine Learning is.

[Slide 5] What is Machine Learning

Machine Learning is the field of study that gives computers the ability to learn without being explicitly programmed.

Let us consider the example of playing the game of checkers. If we were to create a computer game to play Checkers, a traditional approach would be to explicitly program the rules of the game and compute every single possibility that allows the computer to win the game against a human.

But using Machine Learning and AI, we can easily "teach" the system how to play and allow it to practice and learn without computing all combinations.

Formally, ML is formally defined as *[read from the ppt only the second f]*

[Slide 6] ML in Stock Prediction

We can now use ML in Stock Market Prediction and one of the most widely used methods in ML, is **The Support Vector Machine**. SVMs can be used to perform Regression on previous stock data to predict the future closing price any number of days ahead

[Slide 7] EXISTING SYSTEM

In existing systems, the stock prices were predicted by performing time series forecasting, and this was done manually by plotting the prices on a graph on a timeline and making predictions based on the company's performance.

In the proposed system, the prediction is done using a **Sliding-window** method where the dataset is time-sliced and prediction is done for a subset of the data and the window is moved to newer set of data.

A variation of the SVM is the LSSVR and we will talk about these algorithms in the coming slides.

[Slide 8] DISADVANTAGES

The main disadvantage is that the existing method predicts the stock price for a selected set of companies and does not generalise for all markets. They also do not have a graphical user interface which allows direct import of data and so cannot be used by common people.

[Slide 9] PROPOSED SYSTEM

The proposed system generalizes the usage and applies to all major markets over the world. It also includes the development of a user interface that allows importing of live data online because of which the application can be widely used by businessmen who are unfamiliar with the underlying complexity of the program.

[Slide 10,11,12] System Design, Data Flow Diagram, Sequence Diagram

All three slides have the following explanation.

1. This is the working of the system, and the user initially gives the name of the company and date range as input.
2. The app fetches the stock data online and plots them on a graph and displays them
3. The stock data is sent to the LSSVR training algorithm and a model is prepared.
4. The model is used to predict the stock for N days.
5. The predicted stock is plotted on the graph and the model is saved.

[Slide 13] Implementation - ALGORITHM

This is the algorithmic implementation of the application. The inputs are company name, the date range and the number of days to predict and the output will be the predicted stock prices for N days.

1. The data is first fetched for the company within the date range and plotted
2. The sliding window method is applied to the data, which will be explained later
3. For the first day, we feed the stock data to the LSSVM, which prepares a model and predict the stock for the first day.
4. For the second day, we add this predicted result to the existing stock data and feed it back to the LSSVR in the next loop, and we do this for each day.
5. Therefore, for every prediction we make, we feed it back to the algorithm for the next day. We do this for N days and each predicted price is stored and plotted on the graph.

[Slide 14] Implementation – LSSVR Formulation

As stated earlier, the main algorithm used in the proposed system is the LSSVR

The LSSVR is computationally faster because it solves for linear equations. In this method we minimize the cost function J and w is the parameter that fits a non-linear curve to the dataset. (There is so much to this slide, I won't explain more here)

[Slide 15] Implementation – Sliding-Window

Given a sequence of numbers for a time series dataset, we can rearrange the data to look like a supervised learning problem. We can do this by using previous time steps as input variables and use the next time step as the output variable

Time	Price
1	100
2	110
3	108
4	115
5	120

X	Y
?	100
100	110
110	108
108	115
115	120
120	?

We can see that the previous time step is the input (X) and the next time step is the output (y) in our supervised learning problem.

We also have no previous value that we can use to predict the first value in the sequence. We will delete this row as we cannot use it.

Additionally, we see that we do not have a known next value to predict for the last value in the sequence and we shall delete this value while training our supervised model

[Slide 16] RESULTS & DISCUSSIONS

Let us consider a small dataset from the IBM Stock from January 2017 to 2018. To see the working, we shall split the data into two parts.

The blue line represents the past stock data that we use to train the model. We later use the model to predict the future stock [point to the center of the graph] represented in the orange line. The actual stock price on that day is represented in Gray colored line. AS you can see the actual price and the predicted price almost overlap showing the accuracy of the application which is 99.39% for this dataset.

[Slide 17] Performance

We can predict the stock prices of various datasets and plot the accuracy as the program executes. This is the resultant graph. [point to the graph]

[Slide 18] Overall Performance

This is the overall performance of the application. We can conclude that the proposed system gives an accuracy of 98% for up to 60 days after which the accuracy declines.

[Slide 19] Screenshot

[Slide 20] Limitations and Future Enhancement

- The limitation of the proposed system is its computational speed, especially with respect to sliding-window validation as the computational cost increases with the number of forward day predictions.
- The proposed model does not predict well for sudden changes in the trend of stock data.
- This occurs due to external factors and real-world changes affecting the stock market.
- We can overcome this by implementing Sentiment Analysis and Neural Networks to enhance the proposed model.
- We can modify the same system to an online-learning system that adapts in real-time.

[Slide 21] Conclusion

Thus, as we can see above in our proposed method, we train the data using existing stock dataset that is available. We use this data to predict and forecast the stock price of n-days into the future.

[Slide 22] Thankyou

Thika muchkond 300/300 kodu nan-magane, kashtapattu maadidane. *(point at me and tell)*