

# Stock Forecasting Using HMM and SVR

Maturi Tanuj<sup>1</sup>, Aishwarya Virigineni<sup>1</sup>, Apoorva Mani<sup>1</sup> and Subramani R<sup>2</sup>

<sup>1</sup>Department of Computer Science and Engineering, Amrita School of Engineering, Bengaluru, Amrita Vishwa Vidyapeetham, India

<sup>2</sup>Department of Mathematics, Amrita School of Engineering, Bengaluru, Amrita Vishwa Vidyapeetham, India  
[charantanujma63@gmail.com](mailto:charantanujma63@gmail.com)<sup>1</sup>, [aishwaryavirigineni@gmail.com](mailto:aishwaryavirigineni@gmail.com)<sup>1</sup>, [apoorvaa.mani@gmail.com](mailto:apoorvaa.mani@gmail.com)<sup>1</sup>, [r\\_subramani@blr.amrita.edu](mailto:r_subramani@blr.amrita.edu)<sup>2</sup>

**Abstract:** - Stock markets are known for their highly volatile nature which makes it very complex. It changes continuously due to the effect of various attribute, of which many are unknown. This is why it is highly challenging to forecast future stock rates as no particular pattern is followed. However, in order to acquire profits its wise to keep in mind that factors should be analyzed and calculated before investing. Forecasting of stock prices is typical problem of non-stationary pattern recognition in Machine Learning. In the recent times, many methods have been tried and tested in order to forecast the future rates of the stocks by keenly studying the previous rate trends. Hidden Markov model is not a very familiar method that is used in predicting stocks. In this project we'll be using HMM and SVR (comparatively more commonly used in stock forecasting) to predict stocks and compare their performance to deduce which model is more efficient.

**Keywords**—Hidden Markov Model (HMM), Support Vector Regression (SVR), Stock market

## I. INTRODUCTION

Hidden Markov models are statistical models that are very well suited for capturing information that are hidden from the sequence that is observed, typically in areas where pattern recognition plays the main role. The main idea behind Hidden Markov model is that the probability of the data observed is dependent on the unknown (otherwise known as hidden) parameters of the system. Transitioning between the states is defined as the Markov process. This signifies that the coming state is only dependent on the present state and nothing else. Even though the states of the Hidden Markov model are always discrete, the observed data can be either continuous, discrete or both. It was used in Speech recognition firstly and later found its way to be used in various applications related to DNA sequencing. Hidden Markov Model is considered to be Markov model that deals with some parameters that are unknown from the observed data. A Hidden Markov model is considered to be efficient if its able to model the source of observed data and finds success in simulating the same, i.e simulating data of good

accuracy. Hidden Markov Model is now being used many Machine learning algorithms and has been able to solve a wide variety of problems successfully. Especially when it comes to Bioinformatics, Hidden Markov Model has become one of the most fundamentally used tool due to its robustness when it come to statistical foundation. This makes them apt for being used in diverse classification problems. Hidden Markov model's magic applies to Computational Biology too! It is a wonderful method to model biological sequences.

The above applications of Hidden Markov model strongly argues about its efficiency to solve pattern recognition problems. In this project, Hidden Markov model will be used to forecasting the prices of stock market. In this, the stock markets can be seen as the Hidden Markov process out of which the people who are investing can only observe the fluctuation of prices of the stocks and the unknown parameters that affect this fluctuation of rates can be seen as the hidden states. Here the considered observations are the daily open, close, high and low for Tesla stocks. Support vector regression method is one of the most commonly used methods for stock prediction. In this project, both Hidden Markov model as well as support vector regression will be used to forecast stock prices and will later be compared and analyzed based on their performances by considering various criterias.

In recent times, excessive research has been published and is proceeding to discover an ideal (or nearly optimal) forecasting model for the stock market. Most of the predicting research has utilized the statistical time series analysis techniques like autoregression moving average (ARMA) [2] as well as the multiple regression models. In recent years, numerous stock prediction systems based on AI techniques, including artificial neural networks (ANN) [4], LSTM, RNN, CNN [13], support vector regression [5,8] have been proposed. And people have applied HMMs for stock market predictions like In [1], they used fixed state

HMMs for predicting some airline stocks. Fixed state HMM based MAP estimator to maximize the likelihood of observation of all probable sequences is used in [3]. They compared their results with HMM-Fuzzy logic model and Artificial Neural Network. Another approach is to use Support Vector Regression (SVR) which has been implemented in [7]. We have implemented an HMM model similar to that implemented in [9,11] and compared its performance with SVR model [6].

## II. HIDDEN MARKOV MODEL

HMM is a generative stochastic markov model in which the system is considered to determine the probabilistic characteristic of any random process by transitioning in certain number of states. It fundamentally says that an observed event will not be comparing to its step-by-step status but related to a set of likelihood distributions. This state transition is known as a Markov Process and thus a Markov process can be defined using a state transition probability matrix. Let's think of a system which is a Markov chain and within the process, there are few hidden states. In this case, it can be told that hidden states are a process that depends on the main Markov chain. The fundamental objective of Hidden Markov Model is to know about a Markov chain by watching its hidden states.

Hidden Markov Models can be defined using the following terminologies:

- 1) Number of states present in the model
- 2) Number of hidden states and observed states
- 3) State transition probabilities
- 4) Initial state distribution
- 5) Observed emission probability distribution

Hidden Markov Models is a strong tool to analyze non-stationary models. The first most thing that comes under non-stationary models is Stock markets as their observations are continuous in nature.

Let us consider  $O_t$  to be a four-component vector which consists of the four important prices used in stock trading which include Open price, Close price, High price and Low price and  $S_t$  to be the state on day  $t$ . The below figure shows a general Hidden Markov Process.

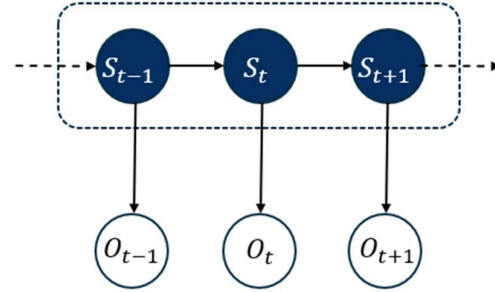


Fig 1: Hidden Markov Model

As the  $O_T$  vector considers real values, observations can be modelled as Multivariate Gaussian distributed. Our observations are considered independent while the observed components might be correlated. As HMM is a finite machine, the state  $S_k$  can consider only discrete values.

We'll be using the following notations regarding HMM in our paper:

$N$  = total number of states,  $(S_T = s_1, s_2, \dots, s_N)$

$T$  = number of observed states

$L$  = latency

$O$  = observed sequences  $(O_1, O_2, \dots, O_T)$

$P_O$  = initial state probability

$A$  = state transition probability matrix =  $[a_{ij}]$ ,  $a_{ij}$  is the state transition probability from  $s_i$  to  $s_j$ .

$\mu_i \Sigma_i$  - Observation Probabilities,  $i = 1, 2, \dots, N$ , where  $\mu_i$ ,  $\Sigma_i$  are the mean and covariance matrix for Gaussian distribution for state  $i$ .

Using the above notations, HMM can be defined as:

$$\lambda = (S, O, A, \mu, \Sigma, P_O)$$

Before working with HMM, the below three questions need to be answered:

- 1) Given the model, what is the probability that a particular sequence of data is produced by that model?
- 2) Given the model  $\lambda$  and sequence of symbols (observations  $O$ ), what is the most likely hidden state sequence that produced the sequence?
- 3) Given the model  $\lambda$ , a set of sequences and the

observation sequence  $O$ , what is the best optimal model that fits the data?

For the first problem, we have used Forward algorithm. The second problem can be solved by Viterbi algorithm. The third problem is solved by Baum-Welch algorithm.

There are few accepted algorithms to answer the above questions. Forward or Backward Algorithms can be used to solve the first problem, The second question can be solved using Viterbi algorithm whereas to resolve the third problem, Baum-Welch algorithm can be used to train HMM.

Stock Market Forecasting using HMM:

The hidden Markov model is broadly used within the financial area to predict economic administrations or anticipate stock pricings.

The foremost thought for foreseeing the next day's stock cost is to figure out the log-likelihood of  $K$  past observations and comparing it with the log-likelihood of all the past sub-sequences of same estimate by moving the window by one day within the course of past information. We at that point recognize a day within the past whose log- probability of its  $K$  past perceptions is the closest to the sub-sequence whose another day's price is to be predicted.

$$j = \underset{i}{\operatorname{argmin}} (|P(O_t, O_{t-1}, O_{t-2}, \dots, O_{t-K} | \lambda) - P(O_{t-i}, O_{t-i-1}, O_{t-i-2}, \dots, O_{t-i-K} | \lambda)|)$$

where  $i = 1, 2, \dots, T/K$

We at that point calculate the differential cost change from the recognized day to its another day. This alter at that point is included to the current day's cost to induce our following day's expectation.

$$O_{t+1} = O_t + (O_{t-j+1} - O_{t-j})$$

Later on, after we get the true observation, we incorporate it to our dataset and retune the parameters of our model by arranging them in order in to guarantee that our model doesn't diverge. In brief, we fix the estimate of our sub-sequence and find another sub-sequence from the past information which shows alike pattern. We at that point outline the behavior of the distinguished sub-sequence to the sub- sequence being utilized for expectation.

### III. SUPPORT VECTOR REGRESSION

SVR is a supervised learning algorithm used for prediction. SVR is almost alike and uses the same principles as Support vector machine. But there is a slight difference, as its name suggests that SVR is a regression algorithm, so we are able to use SVR for working with ongoing Values whereas SVM is being used for classification.

In SVM, when the data and labels are given, the model tries to separate the different classes as much as possible using a hyper plane, thus it predicts what belongs to which class. Whereas in SVR, the basic idea is to find the best fit line within a threshold value. This best fit line is the hyperplane which has the maximum number of points. SVR models generates forecast results depending on model design. Error percentage is computed between the actual stock prices and the forecasted stock price come from trials.

And now for our SVR prediction, we r going to consider Radial Basis Function kernel or RBF kernel to construct our SVR model in the feature space.

Radial Basis Function (RBF) kernel:

RBF is the standard kernel that is used in the sklearn's SVM classification algorithm and it can be said in the following formula:

$$K(x, x') = e^{-\gamma ||x - x'||^2}$$

where gamma can be adjusted manually and has to be greater than zero. Gamma's default value in sklearn's SVM classification algorithm is:

$$\gamma = \frac{1}{n \text{ features} * \sigma^2}$$

Briefly,  $||x - x'||^2$  is a squared Euclidean distance between two feature vectors (2 points). Gamma is a scalar that determines how much influence a single training example (point) has.

So, using gamma values we can control single points' impact on the overall algorithm. Bigger the gamma is, the closer other points must affect the model.

Stock Market forecasting using SVR:

For example, let us consider this dataset which is of the format  $(x_i, y_i)$ .

$$(x_1, y_1), (x_2, y_2), (x_3, y_3), \dots (x_T, y_T)$$

where  $x_i$  denotes all the four prices, i.e., close/open/high/low prices on the day  $i$  and  $y_i$  denotes those prices for the next day i.e.,  $i+1$ .

Using SVR, we create a function where current day's prices from the dataset are considered as input and the fn. tries to predict the next day's prices. So far, SVR is trained with this data. Now to test it, current day's prices are passed as input to SVR, and then we try to predict the next days close/open/high/low prices. And then we finally compare these predicted observations with the true observations. After this again, these true observations are added to our training data to retune our model so that we can make predictions for the day after. And this process goes on and on.

#### IV. IMPLEMENTATION

The performance metric that we used in this project is Mean Absolute Percentage Error (MAPE) which is defined as

$$MAPE = \frac{1}{N} \sum_{i=1}^m \frac{|Predicted(i) - True(i)|}{True(i)}$$

In this paper, our main purpose was to define efficiency of HMMs in forecasting stock prices. We have used hmmlearn, an open-source python library for model training and estimate the probability of the observations. The stocks that we have selected is Tesla Inc. We were using the high price, opening price, closing price and low price as characteristics for beyond 3120 working days (roughly 12 years) when the market is open. We kept apart the recent 30 observations for testing and utilized rest of the observations for training the model. We have forecasted the prices for the last 30 days, starting with the 30th day then using its true observation to reset the forecasting model for 29th day and so on. Therefore, whenever we return the model, the number of training examples will be raised by one. First, we have implemented using fixed model i.e., by determining the number of states to four. Figures xx shows the stock price predictions for Tesla using HMM with four states. We counted MAPE and mapped the forecasts and the actual prices to compare the results.

Figures 2-5 shows the actual and predicted HMM opening, closing, high, low prices of TESLA company for the past 30 days respectively.

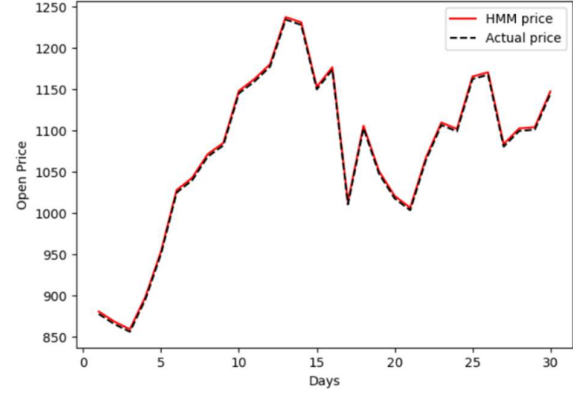


Fig 2: Opening prices of Tesla

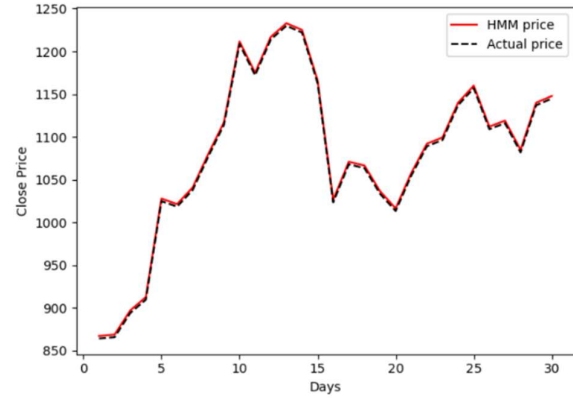


Fig 3: Closing prices of Tesla

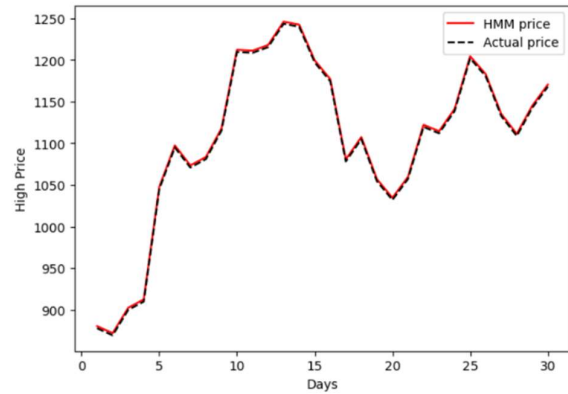


Fig 3: High prices of Tesla

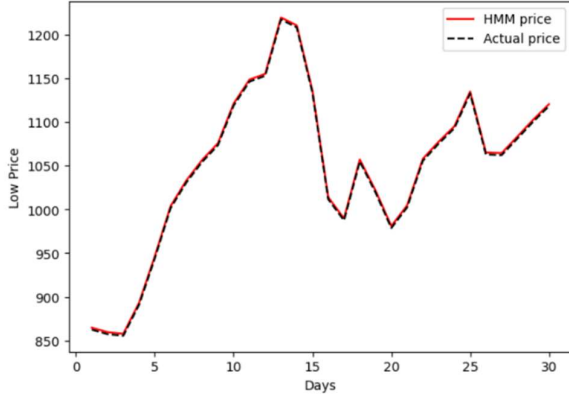


Fig 4: Low prices of Tesla

We have also used a non-probabilistic model- SVR (Support vector regression) to model the stocks. Take a data set  $(x_1, y_1), (x_2, y_2), \dots, (x_T, y_T)$ . In contrast to the Support Vector Machines for classification where  $y_i$  is a label, in SVR  $y_i$  is the answer variable. Using SVR, we tried to find the feature that takes in the current day's rates and forecasts the following day's high/ low/close/open prices. We have created four models to forecast four prices. Testing and training are exactly same as we did for HMM. We start forecasts with 30th day and use its true observation to re-train the SVR predicting for 29th day and so on. Forecast following day prices, we trained the SVR with data from past until last day. Next, we have passed today's prices to get the forecast for tomorrow. We counted MAPE and calculated the predictions on the same plots of HMM to compare results from SVR and HMM. Figures 6-9 shows real as well as forecasted opening/closing/High/low prices for the Tesla stocks using HMM as well as SVR models. Tables 1 shows the MAPE values for all the four stocks values.

## V. RESULTS

We have noticed that the predicted Close, Low, High, Open prices closely take after the patterns exhibited by its corresponding true values in both the SVR and HMM models and the MAPE values were also found to be similar. The forecasts made utilizing the SVR show was not found to be influenced by exceptional changes within the stock cost. In any case, the model executed utilizing HMM was found to be overblown

to the changes in stock price. This result is also consistent with the scatter plot of the prediction error on each day for both the models where the spread of the error points in the case of SVR was observed to be concentrated around zero and the spread for HMM was observed to be more scatter around zero.

Model	Closing	Open	High	Low
HMM	0.2828	0.2834	0.2342	0.2183
SVR	0.3620	2.5224	0.3770	0.3867

Table 1: MAPE values for TESLA

Figures 6-9 shows the scatter plots of the error in the prediction of the Tesla stock using both the HMM and SVR models for each day.

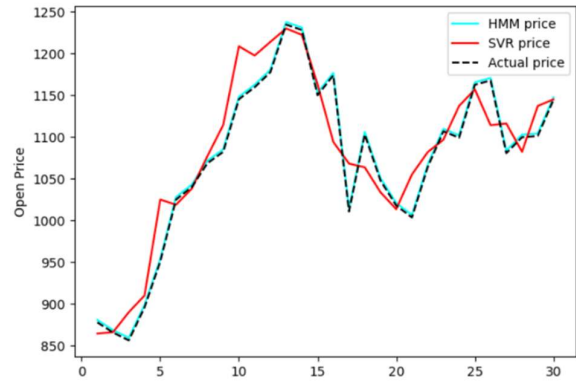


Fig 6: Open price comparison plot

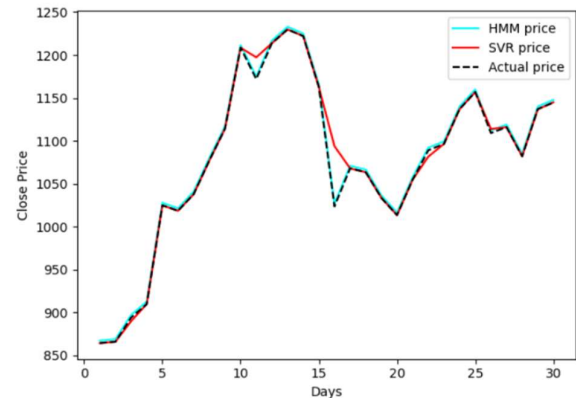


Fig 7: Close price comparison plot

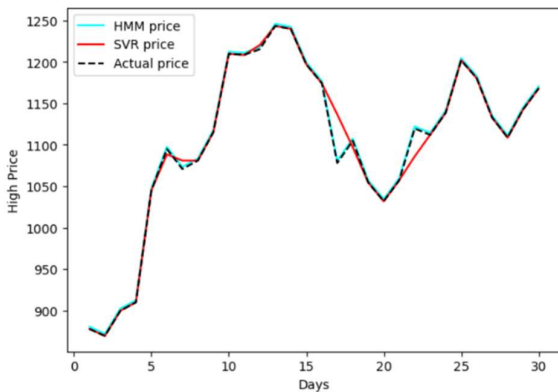


Fig 8: High price comparison plot

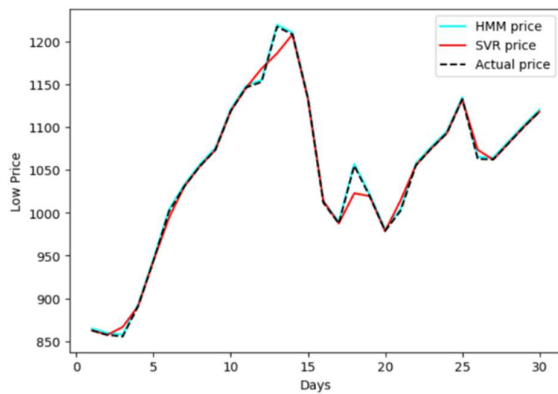


Fig 9: Low price comparison plot

## VI. CONCLUSION

Stock performances are an essential indicator of the strengths and weaknesses of the stock's corporation and the economy in general. In this paper, we have used the hidden Markov model and SVR model to predict monthly closing prices of Tesla company and then used these predictions to trade the stock. Though in general, the observations will be greatly affected by the choice of the model i.e.; the number of states in Hidden Markov Models, but actually it did not make any significant difference when we used HMM for Stocks. Coming to the performance, both HMM and SVR have given similar accuracy when the next one-day prices are predicted. And these predictions diverge when prices are predicted for more than one day. After analyzing the graphs, we came to know that HMM captures the volatility of the stock prices while SVR gives more stable predictions. So, as both the models have their advantages and drawbacks, we can't say a particular model is best for stock prediction.

Instead, what we can say is, HMM can work better for the stocks with high volatility and SVR can work better for stocks which are more stable. Finally, it's not easy task to predict stocks beyond a certain point of accuracy.

## REFERENCES

- [1] Hassan, Md. R.; Nath, B.; Stock Market Forecasting Using Hidden Markov Models: A New approach. Proceeding of IEEE fifth International Conference on Intelligent Systems Design and Applications 2005.
- [2] T. Ye, "Stock forecasting method based on wavelet analysis and ARIMA-SVR model," 2017 3rd International Conference on Information Management (ICIM), 2017, pp. 102-106, doi: 10.1109/INFOMAN.2017.7950355.
- [3] Aditya Gupta, Bhuwan Dhingra. Stock Market Prediction using Hidden Markov Models. IEEE
- [4] Gaurav Kshirsagar, Mohit Chandel, Shantanu Kakade, Rukshad Amaria. Stock Market Prediction using Artificial Neural Networks. International Journal of Advanced Research in Computer Engineering and Technology (IJARCET), Volume 5, Issue 5, May 2016
- [5] Bruno Miranda Henrique, Vinicius Amorim Sobreiro, Herbert Kimura, Stock price prediction using support vector regression on daily and up to the minute prices, The Journal of Finance and Data Science, Volume 4, Issue 3, 2018
- [6] Chi-Yuan Yeh, Chi-Wei Huang, A multiple-kernel support vector regression approach for stock market price forecasting. Expert Systems with Applications, Issue 3, March 2011
- [7] L. J. Cao, Francis E. H. Tay: Support Vector Machine with adaptive parameters in financial time series forecasting. IEEE transactions on neural networks, vol. 14, no. 6, november 2003
- [8] P. Meesad and R. I. Rasel, "Predicting stock market price using support vector regression," 2013 International Conference on Informatics, Electronics and Vision (ICIEV), 2013, pp. 1-6, doi: 10.1109/ICIEV.2013.6572570.
- [9] A. Jadhav, J. Kale, C. Rane, A. Datta, A. Deshpande and D. D. Ambawade, "Forecasting FAANG Stocks using Hidden Markov Model," 2021 6th International Conference for Convergence in Technology (I2CT), 2021, pp. 1-4, doi: 10.1109/I2CT51068.2021.9418216.
- [10] P. V. Chandrika, K. Visalakshmi and K. Sakthi Srinivasan, "Application of Hidden Markov Models in Stock Trading," 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), 2020, doi: 10.1109/ICACCS48705.2020.9074387.
- [11] I. Palupi, B. A. Wahyudi and A. P. Putra, "Implementation of Hidden Markov Model (HMM) to Predict Financial Market Regime," 2021 9th International Conference on Information and Communication Technology (ICoICT), 2021, pp. 639-644, doi: 10.1109/ICoICT52021.2021.9527459.
- [12] V. K. Menon and K. P. Soman, "Stock price prediction using dynamic mode decomposition," 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), doi: 10.1109/ICACCI.2017.8125816.
- [13] S. Selvin, R. Vinayakumar, E. A. Gopalakrishnan, "Stock price prediction using LSTM, RNN and CNN-sliding window model," 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2017, pp. 1643-1647

