Stock Market Values Forecasting

Rohit Singh (12588), Sudhir (12734), Jitendra Singh (12326)
April 20, 2016

INTRODUCTION

Stock values prediction is one of the most important current problem existing on which different algorithms are being tried to improve the accuracy of the result. Stock values simply refers to the valuation of a company at a particular instant of time in the market and this project is to predict the future characteristics values (i.e. closing price,opening price,minimum price, maximum price and average price) of a company .Movement of prices of stock values appears to be random in nature but there can be pattern available in them also.We are predicting the characteristics values by assuming without any relation between them and in the second case assuming a relation between them.

In the first case , we will predict any price (consider closing price) using only closing price values but in the second case , we will take account of all the corresponding prices. At any given instant, the stock values are very much different from their previous values but still we can find hidden relation or pattern in data using Gaussian Processes .

PRIOR WORKS

There are many attempts done on the stock market values prediction using different algorithms especially using ANN and K-nearest neighbours. This prediction has also been tried using Hidden Markov model(HMM). In HMM, some hidden pattern is observed between the sequence provided and according to that prediction is made .In ANN,training using data sets and with single and multiple hidden layers has been tried out .IN this model, weight is updated at each layer and in each iteration according to the input provided.Using KNN, much accuracy was not achieved .In HMM model, from given data set , first cluster assignment probabilities , transition matrix and emission matrix is learned and updated at each iteration and then these

finally learned and updated parameters are used to make prediction. Now, Our algorithm is based on Gaussian Processes and then we have also taken into account of the results from neural network and ARIMA and then comparison is done.

DIFFERENT ALGORITHMS USED

- Gaussian Processes
- ARIMA(Auto-regressive Integrated Moving Average)
- Artificial Neural Networks

Here, we will try different kernels for Gaussian processes namely normal exponential, squared exponential and linear kernel. Now, lets move on to the different algorithms. Gaussian Processes are based on the fact that observations are occuring in the continuous domain and they are assumed to be derived from some normal distribution.

$$y \sim N(0, K(x,x))$$

$$K(x_1, x_2) = exp(-\sigma||x_1 - x_2||^2)$$

$$K(x_1, x_2) = exp(-\sigma||x_1 - x_2||)$$

Predictive distribution

$$\begin{bmatrix} a \\ f(x_*) \end{bmatrix} \sim N\left(0, \begin{bmatrix} k[x,x] & k[x,x_*] \\ k[x_*,x] & k[x_*,x_*] \end{bmatrix}\right)$$

and, from given data set, we learn 'mu' and 'sigma' of the distribution and then according to that for any test data, prediction is done.

ARIMA is generally used for working on time series data.Any time series data provided to it is considered as a combination of original signal function and some noise and this model generally works on the process of filtering the noise from original signal by doing some non linear transformation and taking the differences between consecutive terms up to some level.The average of all errors is termed as moving average.

ARIMA model is defined by 3 parameters(p,d,q) where

- p = number of auto-regressive terms
- d = number of differences to make model stationary
- q = number of lags in forecast errors

ANN(Artificial Neural Network) is based on the working framework of neurons in our body.In this, all inputs are considered as input signals which are modified according to transfer function which is used to give output signal. ANN consists of input layer , output layer and some hidden layers(usually 1 or 2) and some general activation functions used are 'tanh' and logistic function. Activation function produces output on the basis pf different weight assigned to the inputs.

ALGORITHM

Main steps of our algorithm are input data representation and then training the Gaussian Model for our prediction . First we will talk feature/data representation. In the given data-set , we have opening price, closing price, average value , minimum and maximum value of a particular stock at 2300 different times. We have data-set of 2300*5 matrix in which each row correspond to the values at different time while each column correspond to characteristic price at any given time.For training our model , we are using first 2000 data-points while for testing we will use next 300 data points in Gaussian Processes . Complete data is normalized before using it for training.Now, data is converted in the suitable form to be used in the model.Its converted as follows:

- Each row data is converted into matrix form for first training the model and then testing it to get the possible error.
- Matrix is formed of size 10*1490 for input and for each row of matrix the next orginal data is used as label .

Our model is using Gaussian Processes for making prediction while during preprocessing of data filling of the blank values in the dataset and is converted into a form so that we can use data for training our model and then model is tested which is finally used for making predictions. Now, using gaussian processes , we are making prediction in the following way.

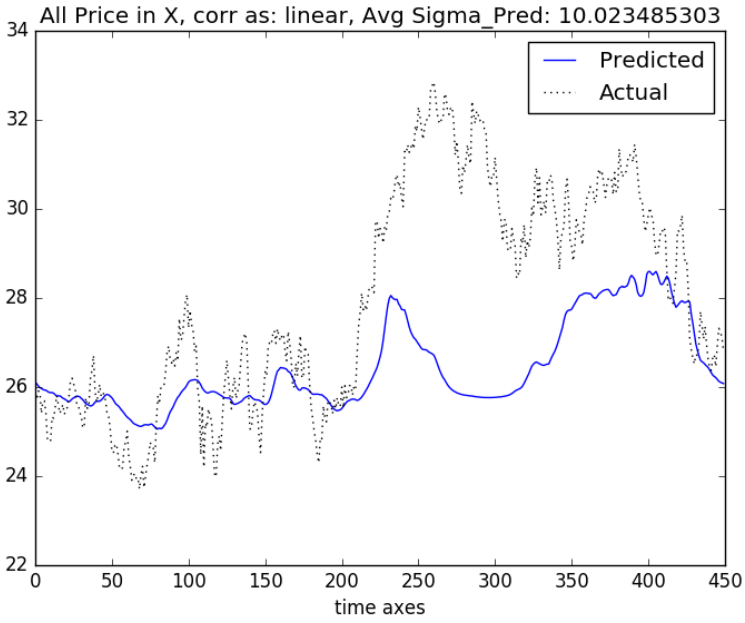
- We are taking 1 to 10 values as input values and 11th data point as prediction.
- Similarly, we take the values from 2nd to 11th in original data point and will take 12th data as label and so on goes for 1500 data points.
- Thus, a matrix of 10×1490 is formed which is used as input for Gaussain Processes and the corresponding label matrix of 1×1490 are used to estimate 'mu' and 'sigma' of the Gaussian and then the rest 800 data points are used for making prediction.
- In the second approach, we are assuming some relation between characteristic prices and thus we are obtaining 40×1490 sized matrix and then again Gaussian Process is trained and prediction.
- Now, both model outputs are compared and the result of better accuracy model is considered.
- Again, the whole process is repeated using different kernels and their results are compared.

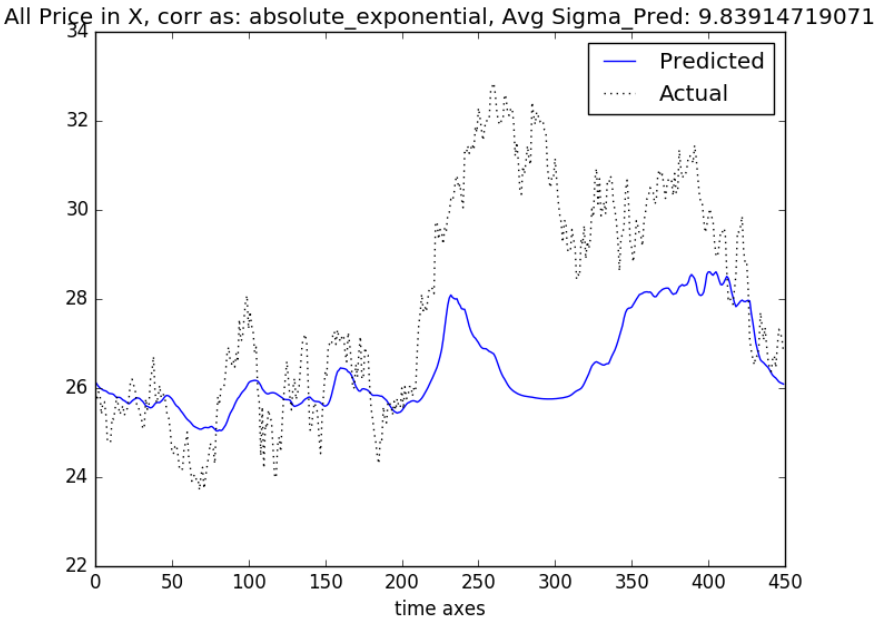
Source Code for experiments can be found here

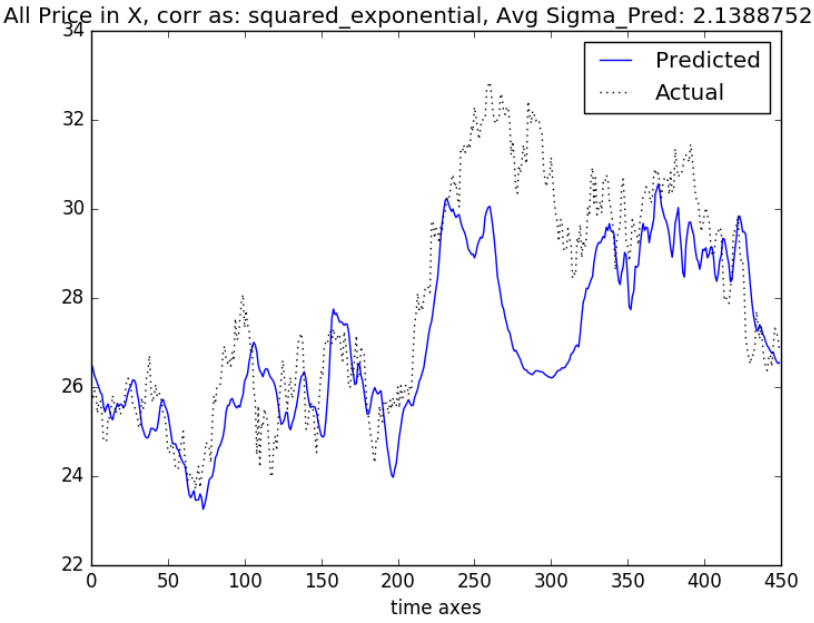
RESULTS

Below are different curves obtained using different kernels and corresponding to each kernel we have also assumed that there is dependence among characteristic prices and in second case we have not assumed any dependency.

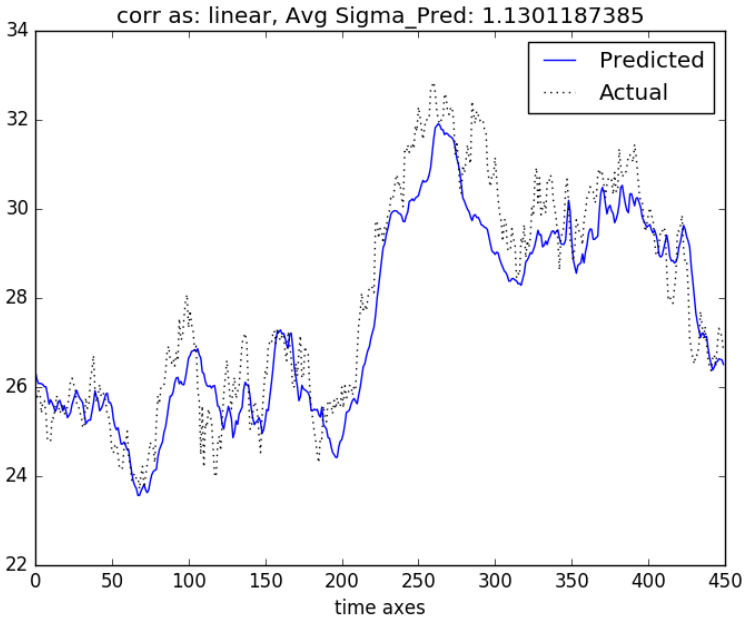
Considering dependency among characteristic prices

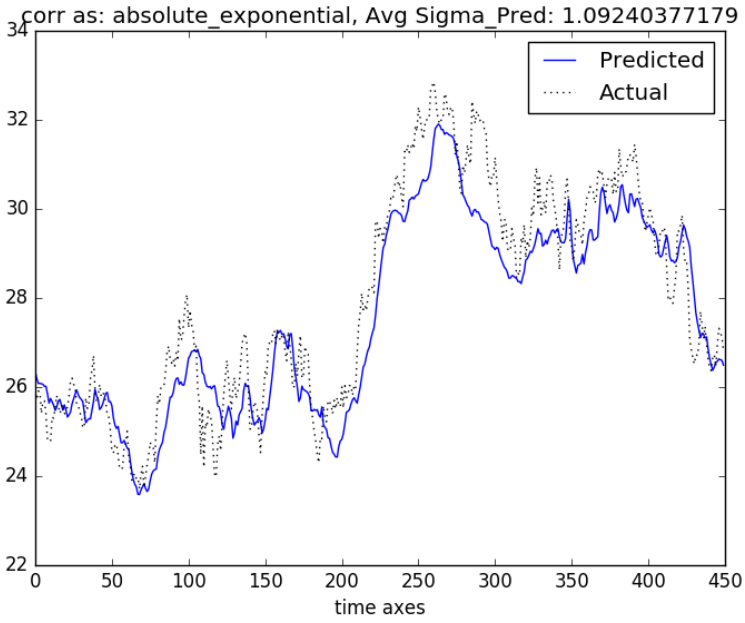


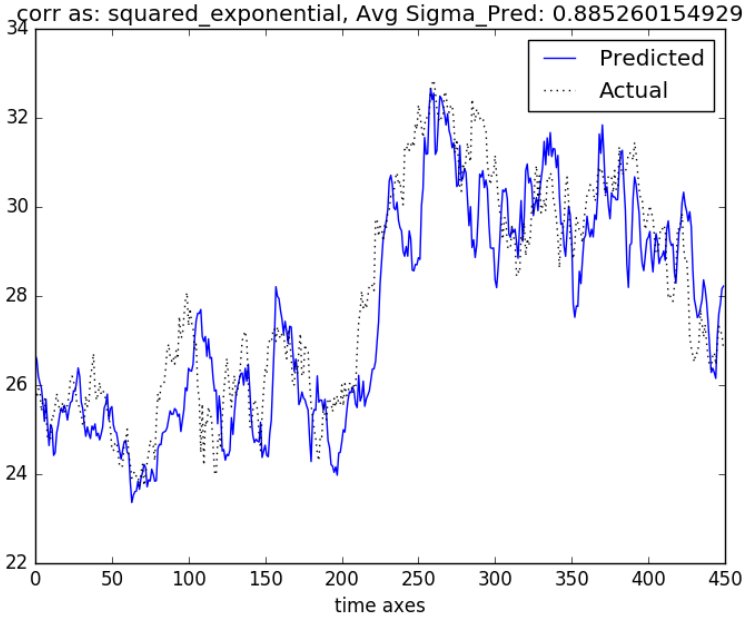




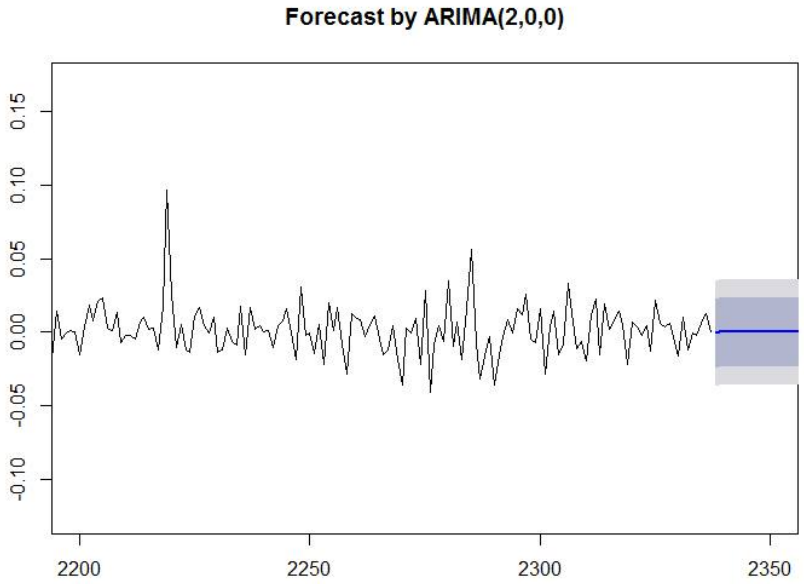
Considering non-dependency among characteristic prices



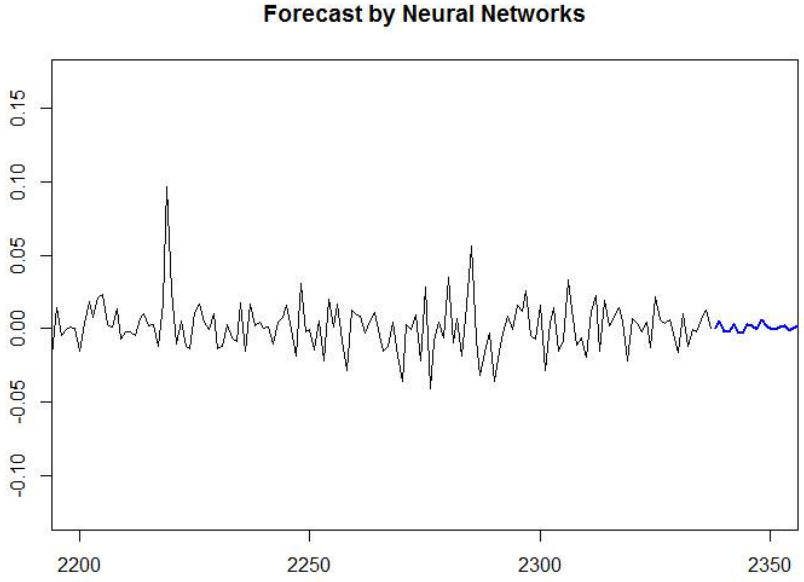

corr as: absolute_exponential, Avg Sigma_Pred: 1.09240377179
Predicted
Actual
time axes


corr as: squared_exponential, Avg Sigma_Pred: 0.885260154929
Predicted
Actual
time axes

ARIMA



ANN



We also did the prediction using ARIMA model and Neural Networks and then we compared their root mean square error.

	ARIMA	NN	GP			GP Correlated		
			abs_exp	sqrt_exp	linear	abs_exp	sqrt_exp	linear
RMSE	0.026%	0.031%	0.892 %	1.177 %	0.906 %	2.545 %	1.851 %	2.561 %

CONCLUSION

From the curves shown above, we can conclude that if we are assuming any dependency among characteristic prices we are getting poor results irrespective of the choice of kernels while among kernels used, we are getting best result from absolute exponential. While in case of non dependency , we are getting best results in case of squared exponential while poorest results are from obtained from using linear kernel.

THINGS WE LEARNED

There are many things we learned during this project.

- Applying Gaussian Processes for a given data .
- Different processes and algorithms already used for stock values prediction.
- Other than this algorithm , we had also tried to use Hidden Markov Model but due to confusion in the number of states and the states themselves , we could not implement that idea.
- Using ANN and ARIMA model on a given time series data.
- We tried K-fold cross validation for better accuracy but result was random .Its because we cannot apply that on time series data.
- We also learned the stock market behaviour as some knowledge in that field was required as prerequisite for this project.

REFERENCES

[1] GPML from Mathworks.
http://in.mathworks.com/help/stats/gaussian-process-regression-models.html?requestedDomain=www.mathworks.com

[2] Stock Market Forecasting.
http://cs229.stanford.edu/proj2012/ShenJiangZhang-StockMarketForecastingusingMachineLearningAlgorith
pdf

[3] http://www.cs.ubc.ca/~nando/540-2013/projects/p5.pdf

[4] https://en.wikipedia.org/wiki/Artificial_neural_network

[5] http://people.duke.edu/~rnau/411arim.htm

[6] Scikit Learn for Python
http://scikit-learn.org/stable/modules/generated/sklearn.gaussian_process.GaussianProcess.html