Stock Market Prediction Using Regression Algorithms

Dr. P.V. Rama Raju[1],
G. Naga Raju[2], K. Surya Mohan[3], K. Naga Harshitha[3],
K. Naga Sai Pujitha[3], Md. Neha Azeemunnisa[3]
[1]Professor, [2]Asst.Professor, [3]B. Tech students
[1,2,3]Department of ECE, SRKR Engineering College (A), Bhimavaram, India.
Corresponding Author mail

Abstract

"STOCK MARKET PREDICTION" is an attempt of trying to predict the future values of a company stock. There are so many factors involved in the prediction of stock market. They are physical, physiological, rational and irrational factors. We can predict the stock value with high accuracy with the help of these factors. In this article, we use the historical data of stock prices of a publicly listed company. Machine Learning algorithms is the best choice in current scenario to such an application. ***.*** *This paper presents some of the machine learning algorithms suitable for its implementation.*

Keywords: *- Machine Learning, Stock Prediction, Supervised Learning, Unsupervised Learning, Linear Regression, Polynomial Regression, Python, Panda and NumPy Libraries.*

1. Introduction:

Analyzing a stock market is categorized into two parts – Fundamental analysis and Technical Analysis.

1.1. Fundamental analysis:

Based on current business environment and financial performance. There are several possible objectives:
a) To conduct a company stock valuation and predict its probable price evaluation.
b) To make a projection on its business performance.

1.2. Technical analysis:

Based on reading charts and using statistical figures. There are two types of approaches:

a) Top Down:

It is a macroeconomic analysis that looks at the overall economy before focusing on individual securities. Traders focusing on this will have short term gain as opposed to long term valuations.

b) Bottom Up:

It focuses on individual stock as opposed to a macroeconomic view. The value in this decision and intend to hold a long-term view on their trades.

Stock Analysis is the method of analyzing the growth of an organization for a certain period. This field is an area of interest not only for investors but also researchers. The main reason for this is due its volatile complex and regular changing nature making it difficult for reliable prediction. Machine Learning is a part of artificial intelligence which provides an ability for a system to automatically learn and improve from experience without coded explicitly. It focuses mainly on the development of computer programs that can access data and use it to learn themselves. The primary aim of this is it allows the computers to learn automatically by not using human assistance and exist actions accordingly. We provide data to the generic algorithms and logic is developed on the basis of that data. When a machine improves its performance based on its past experiences, we can say that machine has learnt truly.

The technique for most accurate prediction is obtained by learning from the past data and to make the program to do this is best possible through machine learning techniques. The decision of buying or selling a stock is based on the mean squared error.

2. Other Related Works:

Dr. P.V. Rama Raju, G. Naga Raju et al. [3] proposed a design which uses the sales force method for prediction of stock value. In this the data is stored in the cloud and used for predict the growth of an organization. H. White, et al. [1] This paper reports some results of an on-going project using neural network modelling and learning techniques to search for and decode nonlinear regularities in asset price movements. We focus here on the case of IBM common stock daily returns. Henry M. K. Mok, et al. [4] developed an approach and was verified by the Granger causality tests, the causality of daily interest rate, exchange rate and stock prices in Hong Kong were explored for the period 1986 to 1991. Depending on the subperiods being considered, sporadic unidirectional causality from closing stock prices to interest rate, and weak bi-directional causality between stock prices and the exchange rate were found. Draper, N. R., Smith, H., & Pownell, E, et al. [12], is an outstanding introduction to the fundamentals of regression analysis-updated and expanded the methods of regression analysis are the most widely used statistical tools for discovering the relationships among variables. This classic text, with its emphasis on clear, thorough presentation of concepts and applications, offers a complete, easily accessible introduction to the fundamentals of regression analysis. Dr. P. V. Rama Raju, G. Naga Raju, et al [18] developed an Artificial intelligence chatbot is predominant these days and getting speed as an application of computer communications. Sometimes chatbot reacts astutely like the human. Lee, S, et al. [8], proposed a model to illustrate the landslide susceptibility mapping, this study applied and verified a Bayesian probability model, a likelihood ratio and statistical model, and logistic regression to Janghung, Korea, using a Geographic Information System (GIS). Landslide locations were identified in the study area from interpretation of IRS satellite imagery and field surveys.

Dong Nguyen Noa hA. Smith Carolyn P.Ros´ e, et al[13],proposed a major thrust of research in sociolinguistics is to understand the connection between the way people use language and their community membership, where community membership can be construed along a variety of dimensions, including age, gender, socioeconomic status and political affiliation. Hal Daum´eIII,et al[19], describes an approach to domain adaptation that is appropriate exactly in the case when one has enough “target” data to do slightly better than just using only “source” data. Our approach is incredibly simple, easy to implement as a preprocessing step. Shlomo Argamon, Moshe Koppel, James Pennebaker, and Jonathan Schlerwith, et. Al [16], proposed a model to find the growth of the blogosphere offers an unprecedented opportunity to study language and how people use it on a large scale. We present an analysis of over 140 million words of English text drawn from the blogosphere, exploring if and how age and gender affect writing style and topic. H.White,et al. [10]proposed an article that learning procedures used to train artificial neural networks are inherently statistical techniques. It follows that statistical theory can provide considerable insight into the properties, advantages, and disadvantages of different network learning methods. We review concepts and analytical results from the literatures of mathematical statistics, econometrics, systems identification, and optimization theory relevant to the analysis of learning in artificial neural networks.

3. Machine Learning Algorithms:

Machine Learning algorithms are classified into following types:

3.1. Supervised Learning Algorithms:

Supervised learning uses labeled training where we get output as y for a given input x based on the mapping. Otherwise, it can be explained as the following equation:

$Y = f(X)$.It allows us to get accurate output for a given input.

Learning is categorized into two types: classification and regression.

We use classification when the output variable is in the form of categories. A classification model looks at the input data and tries to predict labels.

Regression is used to predict the output when the output is real for a given sample. For example, a regression model is used to predict rainfall, height of a person based on the input data. “Ensembling” is another type of supervised learning. It is also known as combining the predictions of multiple machine learning models which are weak when taken individually which gives an accurate prediction on a new data.

3.2. Unsupervised Learning Algorithms:

Unsupervised learning models is used when there are no corresponding output variables for a given input data. It uses unlabeled data to find out the underlying structure present in the data.

There are three types of unsupervised learning:

Association is used to find the probability of occurrence of an item in the list. It is extensively used in market-basket analysis. For example, association is used to predict if a customer buys rice, user is 70% likely to buy dhal.

Clustering is defined as the grouping the objects of similar type in a cluster and the objects in a cluster are different from the objects of another cluster.

Dimensionality Reduction is used to reduce the count of variables in the data set without the information loss. Reduction in dimensions can be done using the following two methods:
Feature Extraction methods and Feature Selection methods. Feature Selection is used to select the subset from the original variables. Feature Extraction is a method which is used to transform data from high end to low end.
Example: PCA algorithm is a Feature Extraction approach.

3.3. Reinforcement learning:

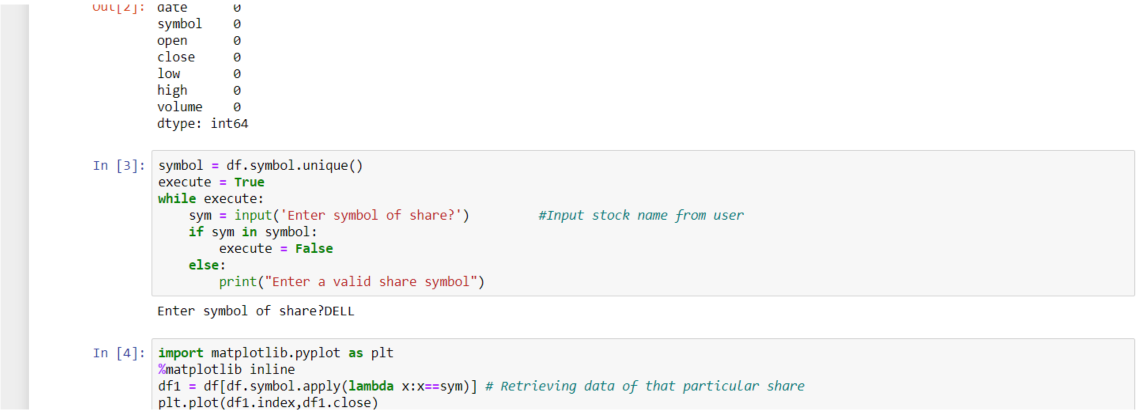
Reinforcement learning is another type of learning in which a user decides the next action based on his present state such that user gets a maximize reward.

Reinforcement algorithms learns by using trial and error method. Imagine, a video game player who needs to move to certain places to score more points. A reinforcement algorithm will start randomly at any point and over through trial and error, it will come to know through which position it has to move to gain maximum points.

4. Methodology:

In this we predict the stock prices of a stock for required number of days in the future based on the Close price. We import required dependencies, that will allow the program a little easier. We import the machine learning library quandl, and NumPy.

Next, we get the stock data of a particular company from quandl, and take a look on the data set. Here we allow the user to select a particular stock data of a company using company keyword like HP, DELL as shown in Figure 1., and store it into a variable ‘df’ which is short for data frame, and printing the first 5 rows of data.

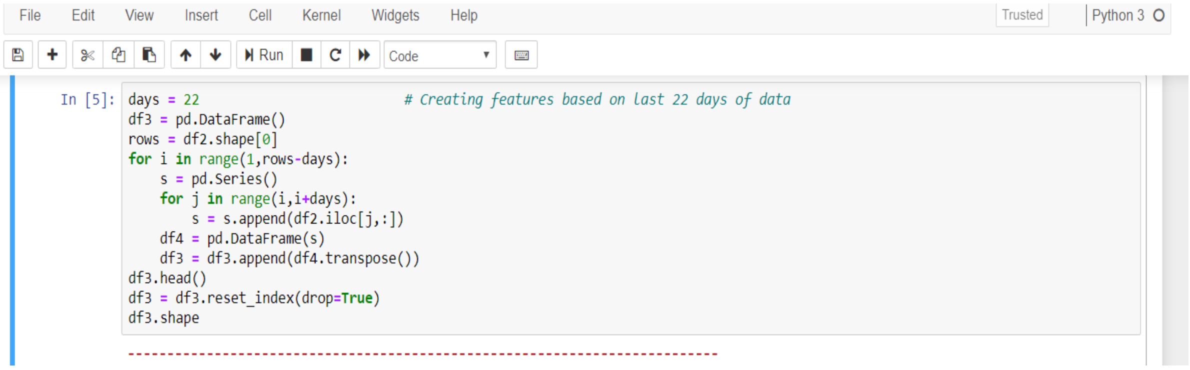


```
Out[2]: date      0
        symbol    0
        open      0
        close     0
        low       0
        high      0
        volume    0
        dtype: int64

In [3]: symbol = df.symbol.unique()
        execute = True
        while execute:
            sym = input('Enter symbol of share?')        #Input stock name from user
            if sym in symbol:
                execute = False
            else:
                print("Enter a valid share symbol")

        Enter symbol of share?DELL

In [4]: import matplotlib.pyplot as plt
        %matplotlib inline
        df1 = df[df.symbol.apply(lambda x:x==sym)] # Retrieving data of that particular share
        plt.plot(df1.index,df1.close)
```

Figure 1. Showing the way to enter company user need

We only need the Close price, so we get data only from the column ‘Close’ and store this back into variable ‘df’. Then we print the first 5 rows of data set we stored back into the variable ‘df’. We get first 5 rows of the data which has Close price column.

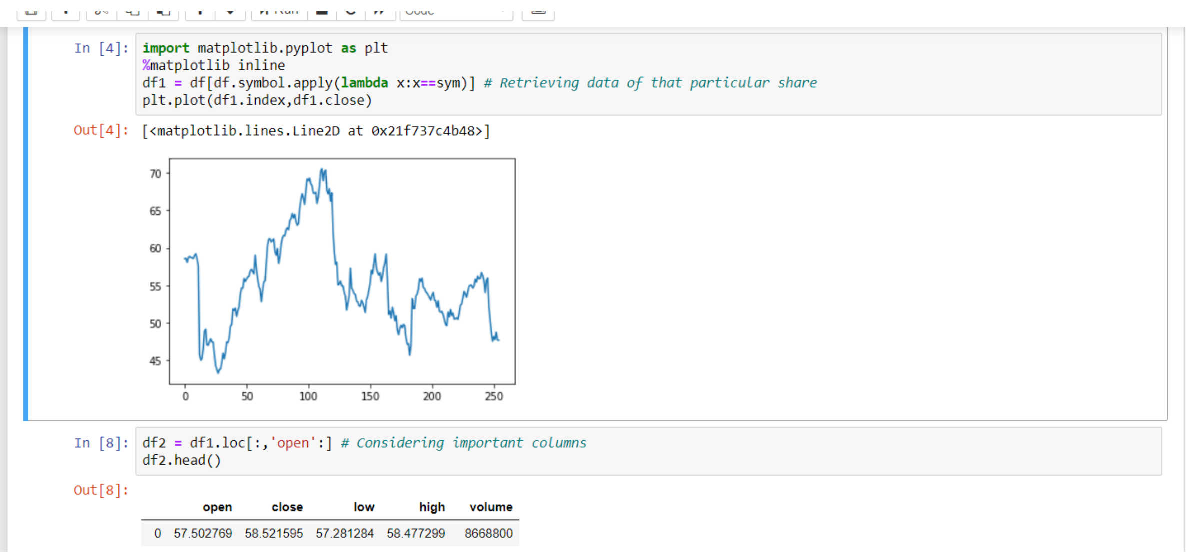
Now, we create a variable to store the predict values, to store the number of days we required in the future we would like to predict. This variable is used in whole programming such a way that we simply change the number and rest of program will be same. So, if user decides the number of days user required to predict as shown in Figure 2. For instance, user requires for 20 days into the future, user changes the variable from 30 to 20, and now program will predict future values up to 20 days.



```
In [5]: days = 22                         # Creating features based on last 22 days of data
        df3 = pd.DataFrame()
        rows = df2.shape[0]
        for i in range(1,rows-days):
            s = pd.Series()
            for j in range(i,i+days):
                s = s.append(df2.iloc[j,:])
            df4 = pd.DataFrame(s)
            df3 = df3.append(df4.transpose())
        df3.head()
        df3 = df3.reset_index(drop=True)
        df3.shape
```

Figure 2. Shows the number of user needs.

Now, we need a column which holds the values of the stock which are predicted for next required number of days into the future. The future price that we want 30 days into the future is just 30 rows down from the Close price. A new column named Prediction is created and it is then used to store the prices for next required number of days by shifting the column up. We plot the shares we got of a particular share as shown in figure 3. Note: As we shifted the data by required number of days those spaces in the prediction column will have NaN’s.



```
In [4]: import matplotlib.pyplot as plt
        %matplotlib inline
        df1 = df[df.symbol.apply(lambda x:x==sym)] # Retrieving data of that particular share
        plt.plot(df1.index,df1.close)

Out[4]: [<matplotlib.lines.Line2D at 0x21f737c4b48>]
```

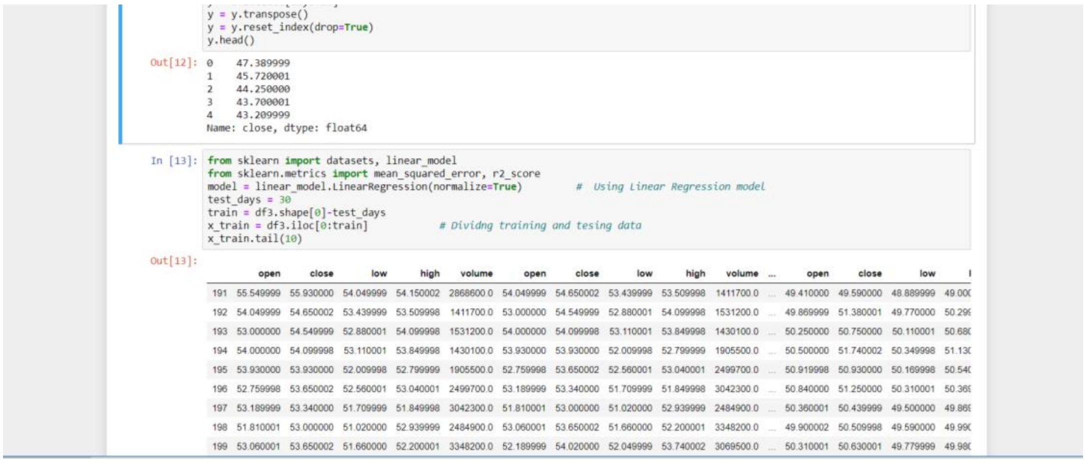
```
In [8]: df2 = df1.loc[:,'open':] # Considering important columns
        df2.head()

Out[8]:
```

	open	close	low	high	volume
0	57.502769	58.521595	57.281284	58.477299	8668800

Figure 3. Plotting the data of a particular share.

We add the new data set after addition of column which is used to store the prediction value by shifting the data up by required number of days. Next, we need to create the independent data set (X). This data set is used to train the machine learning models. For this we do create a variable 'X', and convert all the data into a NumPy (np) array. This conversion takes after adding the 'Prediction' column, and this new data is stored into 'X'. Then we remove the last rows which count is equal to the number of days we like to predict of data from 'X', and then store the new data again into 'X'. We print the data

The new independent data set 'X' we created previously, now we create a dependent data set called 'y'. This is what we called target data which holds the future price predictions.
In order to create this new data set 'y', we need to convert it from data frame to NumPy array and from the 'Prediction' column, and store this into a new variable 'y' and we remove the last columns and the column count is equal to number of days we like to predict .Now, we print 'y' to ensure that there are no NaN's. Now that we have our new cleaned and processed data sets named 'X' & 'y'. we split them up into two sets where 80% of data is used for training and 20 % of data is used for testing model(s) as shown in Figure 4. Next, we create & train the Linear Regression model with the data of the company we chose.

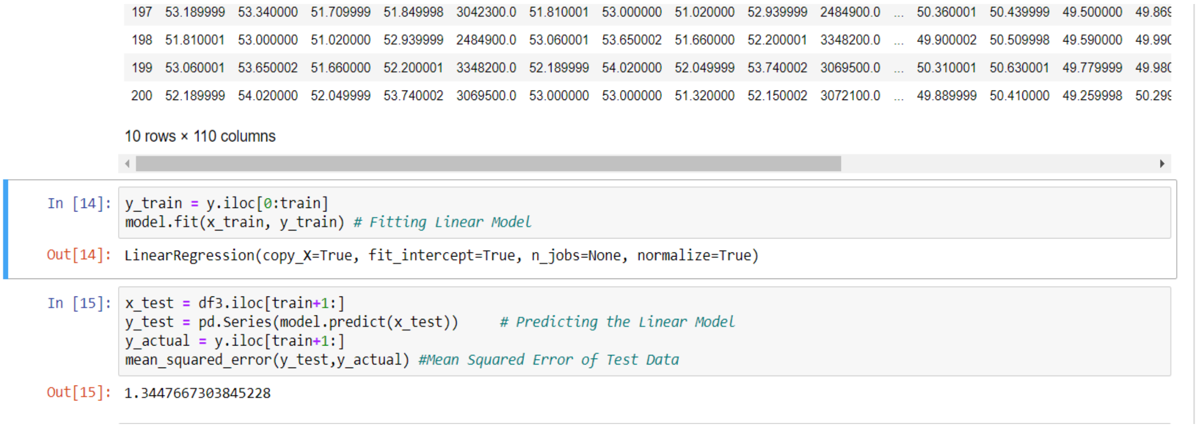


```
        y = y.transpose()
        y = y.reset_index(drop=True)
        y.head()

Out[12]: 0    47.389999
         1    45.720001
         2    44.250000
         3    43.700001
         4    43.209999
         Name: close, dtype: float64

In [13]: from sklearn import datasets, linear_model
         from sklearn.metrics import mean_squared_error, r2_score
         model = linear_model.LinearRegression(normalize=True)          #  Using Linear Regression model
         test_days = 30
         train = df3.shape[0]-test_days
         x_train = df3.iloc[0:train]               # Dividng training and tesing data
         x_train.tail(10)

Out[13]:
```

	open	close	low	high	volume	open	close	low	high	volume	...	open	close	low	h
191	55.549999	55.930000	54.049999	54.150002	2868600.0	54.049999	54.650002	53.439999	53.509998	1411700.0	...	49.410000	49.590000	48.889999	49.00
192	54.049999	54.650002	53.439999	53.509998	1411700.0	53.000000	54.549999	52.880001	54.099998	1531200.0	...	49.869999	51.380001	49.770000	50.29
193	53.000000	54.549999	52.880001	54.099998	1531200.0	54.000000	54.099998	53.110001	53.849998	1430100.0	...	50.250000	50.750000	50.110001	50.68
194	54.000000	54.099998	53.110001	53.849998	1430100.0	53.930000	53.930000	52.009998	52.799999	1905500.0	...	50.500000	51.740002	50.349998	51.13
195	53.930000	53.930000	52.009998	52.799999	1905500.0	52.759998	53.650002	52.560001	53.040001	2499700.0	...	50.919998	50.930000	50.169998	50.54
196	52.759998	53.650002	52.560001	53.040001	2499700.0	53.189999	53.340000	51.709999	51.849998	3042300.0	...	50.840000	51.250000	50.310001	50.36
197	53.189999	53.340000	51.709999	51.849998	3042300.0	51.810001	53.000000	51.020000	52.939999	2484900.0	...	50.360001	50.439999	49.500000	49.86
198	51.810001	53.000000	51.020000	52.939999	2484900.0	53.060001	53.650002	51.660000	52.200001	3348200.0	...	49.900002	50.509998	49.590000	49.99
199	53.060001	53.650002	51.660000	52.200001	3348200.0	52.189999	54.020000	52.049999	53.740002	3069500.0	...	50.310001	50.630001	49.779999	49.98

Figure 4. Dividing training and testing data

The way to test the model is by getting the score which is also known as the coefficient of determination R² of the prediction as shown in figurer 5. The best possible score is 1.0. Based on the R^2 value we go with the value whether to go with which type of regression.



197	53.189999	53.340000	51.709999	51.849998	3042300.0	51.810001	53.000000	51.020000	52.939999	2484900.0	...	50.360001	50.439999	49.500000	49.86
198	51.810001	53.000000	51.020000	52.939999	2484900.0	53.060001	53.650002	51.660000	52.200001	3348200.0	...	49.900002	50.509998	49.590000	49.99
199	53.060001	53.650002	51.660000	52.200001	3348200.0	52.189999	54.020000	52.049999	53.740002	3069500.0	...	50.310001	50.630001	49.779999	49.98
200	52.189999	54.020000	52.049999	53.740002	3069500.0	53.000000	53.000000	51.320000	52.150002	3072100.0	...	49.889999	50.410000	49.259998	50.29

10 rows × 110 columns

```
In [14]: y_train = y.iloc[0:train]
         model.fit(x_train, y_train) # Fitting Linear Model

Out[14]: LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None, normalize=True)

In [15]: x_test = df3.iloc[train+1:]
         y_test = pd.Series(model.predict(x_test))     # Predicting the Linear Model
         y_actual = y.iloc[train+1:]
         mean_squared_error(y_test,y_actual) #Mean Squared Error of Test Data

Out[15]: 1.3447667303845228
```

Figure 5. Mean squared error value

Now we are ready to predict the values now, we consider the last required number of rows of data from the Close price data frame, and store it into a variable called x_forecast after transforming it into a NumPy array and dropping the ‘Prediction’ column of course. Finally, we arrived at the moment and print out the values for next required number of days using the linear regression model. The same procedure is followed for the polynomial regression also.

5. Conclusion:

We discussed about the two widely used methods that are used earlier. Later machine learning methods were applied on a large scale of data. Results of some methods have given a hope but went in vain when checked through realistic simulations. From this, we can conclude that that there is difference in theory and practice of stock market. This report shows the prediction of stock market in extremely tough situation.

References:

[1]. H. White, “Economic prediction using neural networks: the case of IBM daily stock returns,” In Proceedings of the second IEEE annual conference on neural networks., II, pp. 451– 458,1988.

[2]. Min, Jae H., and Chulwoo Jeong. 2009. A binary classification method for bankruptcy prediction, Expert Systems with Applications 36(3), 5256-5263.

[3]. Dr. P.V. Rama Raju, G. Naga Raju, “Analysis of Stock by using sales force methodology”, International Journal of Research, volume 8(4).

[4]. Henry M. K. Mok, “Causality of interest rate, exchange rate and stock prices at stock market open and close in Hong Kong,” Asia Pacific Journal of Management, Vol. 10 (2), pp. 123–143,1993.

[5]. W.C. Chiang, T. L. Urban and G. W. Baldridge, “A neural network approach to mutual fund net asset value forecasting,” Omega International Journal of Management Science., Vol. 24 (2), pp. 205–215,1996.

[6]. K.J. Kim and I. Han, “Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index, Expert Systems with Applications,” Vol.19.

[7]. Y. Romahi and Q. Shen, “Dynamic financial forecasting with automatically induced fuzzy associations,” In Proceedings of the 9th international conference on fuzzy systems., pp. 493–498,2000.

[8]. Lee, S. 2004. Application of likelihood ratio and logistic regression models to landslide susceptibility mapping using GIS. Environmental Management 34(2), 223-232.

[9]. Öğüt, Hulisi, et al. 2009, Detecting stock-price manipulation in an emerging market: The case of Turkey, Expert Systems with Applications 36(9), 11944-11949.

[10].H.White, Learning in artificial neural networks: a statistical perspective, Neural Computation., Vol. 1 , pp. 425–464,1989.

[11].Gharehchopogh, F. S., &Khalifehlou, Z. A. (2012). A New Approach in Software Cost Estimation Using Regression Based Classifier. AWER Procedia Information Technology and Computer Science, Vol: 2, pp. 252-256.

[12].Draper, N. R., Smith, H., &Pownell, E. (1966). Applied regression analysis (Vol. 3). New York: Wiley.

[13].DongNguyen, NoahA.Smith, CarolynP.Ros´e, Author Age Prediction from Text using Linear Regression.

[14].Galen Andrew and Jianfeng Gao. 2007. Scalable training of l1-regularized log-linear models. In Proc. of ICML.

[15].ShlomoArgamon, Moshe Koppel, Jonathan Fine, and Anat R. Shimoni. 2003. Gender, genre, and writing style in formal written texts. Text,23(3):321–346.

[16].Shlomo Argamon, Moshe Koppel, James Pennebaker, and Jonathan Schler. 2007.Mining the blogosphere: age, gender and the varieties of self-expression.

[17].Federica Barbieri. 2008. Patterns of age-based linguistic variation in American English. Journal of Sociolinguistics,12(1):58–88.

[18].Dr. P. V. Rama Raju, G. Naga Raju, "Implementation of chatbot using artificial Intelligence and Natural Learning Processing", International Journal for Innovative Engineering and Management Research, volume 7(4).

[19].Hal Daum´e III. 2007. Frustratingly easy domain adaptation. InProc.of ACL.

[20].Aczel, A. D., 1989. Complete Business Statistics. Irwin, p. 1056. ISBN 0-256-05710-8.