

Stock Price Prediction using Machine Learning

Project report submitted in fulfillment of the requirement for the degree
of Bachelor of Technology

in

Computer Science and Engineering/Information Technology

By

Sanya Ahuja(161341)

Under the supervision of

(Dr. Vivek Sehgal)

to



Department of Computer Science & Engineering and Information
Technology

**Jaypee University of Information Technology, Wagnaghat, Solan-
173234, Himachal Pradesh**

Candidate's Declaration

I hereby declare that the work presented in this report entitled “ **Stock Price Prediction Using Machine Learning**” in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology in Computer Science and Engineering/Information Technology** submitted in the Department of Computer Science & Engineering and Information Technology, Jaypee University of Information Technology, Waknaghat is an authentic record of my own work carried out over a period from August 2019 to December 2019 under the supervision of **Dr. Vivek Sehgal** (Associate Professor, Computer Science).

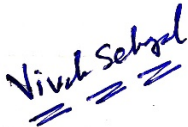
The matter embodied in the report has not been submitted for the award of any other degree or diploma.



Sanya Ahuja

161341

This is to certify that the above statement made by the candidate is true to the best of my knowledge.



Dr. Vivek Sehgal
Associate Professor
Computer Science
Dated: 25-07-2020

ACKNOWLEDGEMENT

I am highly indebted to our supervisor Dr. Vivek Sehgal for his guidance and constant supervision as well as providing necessary information regarding the project and also for his support in completing the project.

We would like to express our gratitude towards HOD of our department Dr. Samir Dev Gupta, for his kind co-operation and encouragement which helped us in completion of this project and for giving us such attention and time.

Our thanks and appreciation also goes to the faculty of Computer Science Department of Jaypee University of Information Technology, Waknaghat for their constant support and motivation.

LIST OF FIGURES

Figure Number	Title	Page Number
1.	SVM Diagram	18
2.	Classification of data	20
3.	Proposed Algorithm for LR	25
4.	Steps in Decision Tree	26
5.	Representation of Decision Tree	27
6.	Flowchart for Proposed System	27
7.	Flowchart for System Architecture	28
8.	Model Design	29
9.	Training Snippets	33-36
10.	Testing Snippets	37
11.	Output-1	41
12.	Error Term for LR	41
13.	Error Term for SVM	43
14.	Error Term for DT	44

LIST OF GRAPHERS

Graph Number	Title	Page Number
1.	Actual and Predicted Y for Linear Regression	41
2.	Stock Price Values v/s Training examples for Linear Regression	42
3.	Profit v/s Values for Linear Regression	42
4.	Pie Chart Representation for Linear Regression	42
5.	Actual and Predicted Y for Support Vector Machine	43
6.	Stock Price Values v/s Training examples for Support Vector Machine	43
7.	Profit v/s Values for Support Vector Machine	44
8.	Pie Chart Representation for Support Vector Machine	44
9.	Actual and Predicted Y for Decision Tree	45
10.	Stock Price Values v/s Training examples for Decision Tree	45
11.	Profit v/s Values for Decision Tree	46
12.	Pie Chart Representation for Decision Tree	46

LIST OF ABBREVIATIONS

Abbreviation	Full form
SVM	Support Vector Machine
API	Application Programming interface
LR	Linear Regression
BSE	Bombay Stock Exchange
NSE	National Stock Exchange
SVR	Support Vector Regression
LSTM	Long Short Term Memory
RNN	Recurrent Neural Network
CSV	Comma Separated Values
DT	Decision Tree

Abstract

In the world of finance, stock trading is the most essential activity. Predicting the stock market is an act of determining the value of a stock in near future and other financial instruments traded on the financial exchange such as NSE, BSE. The fundamental and technical analysis is being in use by the brokers of stock exchange when stocks are being predicted. Here in this report we proposed the method which is called as Machine learning (ML) which is made available by training the stock data, will then gain intelligence and thus finally uses the acquired knowledge for an appropriate prediction. We used many techniques such as Linear Regression, Support Vector Machine and Decision Tree to predict prices of a stock for small and large capitalizations and in the different markets, employing prices daily with the minute frequencies. Linear Regression is used for when the data is in the form of Linearity, or the data seems to be nearby the line to get fitted. In Support Vector Machine, when the data is spread then the line from where the most of the points pass is drawn and from there the vectors from the points to the line are drawn. Meanwhile, in Decision Tree based on the previous data decisions are made that effect of all the alternatives are checked and the most suitable one is decided for the work to be performed.

Table of Contents

Chapter-1	1
INTRODUCTION	1
1.1 Overview	1
1.2 Problem Statement	1
1.3 Objective	1
1.4 Methodologies	2
1.4.1. Data Pre-Processing.....	2
1.4.2. Feature Selection and Feature Generation	2
1.5 Organization	2
Chapter-2	4
LITERATURE SURVEY	4
2.1 Summary of Papers.....	4
Chapter-3	16
System Development.....	16
3.1 Approach using Support Vector Machine (SVM)	16
3.2 Classification and Regression	17
3.2.1 Classification SVM Types	18
3.2.2 Regression SVM.....	20
3.3 Abstraction based extraction.....	20
3.4 Parameters	22
3.5 Approach Using Linear Regression	22
3.5.1 Interpretation	22
3.5.2. Proposed Algorithm.....	23
3.6 Approach using Decision Tree Analysis.....	23
3.7 Proposed System for Extractive approach	25
3.8 System architecture for Extractive approach	26
3.9 Model Design	27
Chapter-4	28
Performance Analysis.....	28
4.1 Proposed solutions	28
4.2 Analysis	29
4.2.1 Prominent features based analysis	29
4.2.2 Prediction analysis	30
4.3 Performance Measures.....	30
4.4 Training Dataset	31
4.5 Testing Snippet.....	35
4.6 Custom Input	35
Chapter 5	45
Conclusion.....	45
5.1 Future scope of improvement	46
References	47

Chapter-1

INTRODUCTION

This chapter is made to introduce you with the . A basic and rough idea of what is the aim and problem statement of our project. We have also mentioned what are we trying to accomplish in this project. Also, all the technologies and platforms have been listed below in the project.

1.1 Overview

Securities exchange expectation is utilized to anticipate the future estimations of organizations stock or other money related instruments that are completely showcased on monetary trades. Notwithstanding, the financial exchange is affected by numerous elements, for example, political occasions, monetary conditions and brokers desires. Be that as it may, Stock market changes are totally irregular but on the other hand are explicit.

1.2 Problem Statement

The point of the task is to ascertain or anticipate the future stock costs of organizations utilizing an alternate number of AI and estimating strategies reliant on authentic returns just as numerical news markers to fabricate an arrangement of numerous or different stocks so as to expand the issue. We do this by putting managed learning techniques for stock value anticipating by understanding the idea of dataset.

1.3 Objective

To create take a dataset of renowned company.

Feature extraction using fundamental analysis

Applying reduced dataset

Evaluating accuracy

Plotting and analyzing the graph

1.4 Methodologies

1.4.1. Data Pre-Processing

The pre-processing stage formally involves

- a. Data discretization: Reducing the data but with our importance and in accordance with the algorithms
- b. Data transformation: Normalizing the data
- c. Data Cleaning: Cleaning i.e. removing all the unnecessary elements from the dataset keeping the ones we actually needed.
- d. Data Integration: Integration of data files

After the data-set is transformed into clean data-set, training and testing datasets are being taken from the dataset for further work and evaluation. Here, the training values are taken as the more recent values because we focus more on training the algorithms and model first. We have kept testing dataset as five to ten percent of the training dataset.

1.4.2. Feature Selection and Feature Generation

We created new features which will eventually provide the better insights of the project like calculating mean, standard deviation, difference in prices, mean errors, squared errors, r2 score etc. We select properties as per the SVM regression, Linear regression and Decision Tree with help of model which is linear for testing the single regressor effects or for many regressors sequentially. We used the Support vector regression, Linear Regression, Decision Tree algorithm in our project.

1.5 Organization

1st Chapter: It includes briefing of the project. A fundamental idea of what is the target of our project and the problem statement. We had also mentioned what we are trying to complete in this project. In addition to, all the technologies and platforms have been listed below in the project.

2nd Chapter: Second chapter contains all the literature survey. We have written about all the research papers that we have studied throughout in understanding and developing of the project.

a variety of papers and publications from well-known sources on machine learning and neural network have been stated in this unit.

3rd Chapter: System development is being covered in this unit. We have mentioned how the model and our project have been evolved over the time. We have drawn several flow charts for model extraction and system extraction for better understanding of the project.

4th Chapter: Fourth chapter includes all the algorithms and mathematical formulas used in our project. Different steps used in applying algorithms. Inputs and their Outcomes have been covered, which are then analyzed.

5th Chapter: Fifth chapter discussed conclusion of the outputs that on what basis it came out , and future scope of the project have been written.

Chapter-2

LITERATURE SURVEY

This section contains ten Research papers and journals that we have studied from various reputed sources.

2.1 Summary of Papers

2.1.1

<i>Title</i>	A forecast approach for securities exchange instability dependent on time arrangement information [1]
<i>Authors</i>	Mohmadd khan M. Asraaafi Alam Parul Goel
<i>Year of Publications</i>	25 January 2019
<i>Publishing Details</i>	IEEE Access
<i>Summary</i>	This paper tells why not the notion of series of time analysis and forecasting should be correct in the respect of Indian economy. The main fall of the currency in the previous times had lead to the important need. The paper not only tries to make an efficient Model but also to guess the Indian stock market volatility. The available always time series data of Indian stock market has been used for this study. The analysed time series has been compared with the original time series, which shows roughly a deviation of 5% mean percentage mistake for both Nifty and Sensex on average. Different tests can be tried for the validation of the predicted time series.

2.1.2

<i>Title</i>	Near investigation of stock pattern expectation utilizing time delay, intermittent and probabilistic unbiased systems [2]
<i>Authors</i>	Saad Prokhorov V.D D.C. Wunsch
<i>Year of Publications</i>	Nov 1998
<i>Publishing Details</i>	IEEE Transactions on Neural Networks
<i>Summary</i>	<p>Foreseeing condensed term stock patterns are reliant on history or past estimations of day by day shutting costs which are conceivable utilizing any of the various systems talked about. At first assurance examination and neuro-designing are significant for fruitful execution. The correlation coefficient measured is important in estimating the delay between the inputs of TDN. A minute exponent indicates either a cyclic or systematic behavior of the stock. Later here, it was noticed that a training set which is relatively short should be used. TDNN is moderate in terms of memory requirement and implementation complexity. PNN has advantages of extreme implementation which should be simple and low false alarm rate even for stocks with low predictability. PNN is more suitable for stocks which do not need training on long or large history, foreign stocks of Apple. Like TDNN, RNN don't need large storage memory, but the wrong thing is the complexity of implementation—a one-time task.</p>

--	--

2.1.3

<i>Title</i>	Forward figure of stock value utilizing sliding window metaheuristic-improved AI relapse [3]
<i>Authors</i>	Sheng Chou Kha Nguyen
<i>Year Publications of</i>	July 2018
<i>Publishing Details</i>	IEEE Transactions on Industrial Informatics
<i>Summary</i>	Choice to sell or purchase a stock is perplexing since numerous components are influenced by stock cost. This presents a novel methodology, in view of a MetaFA-LSSVR, to develop an anticipating of stock value master framework, with the point of improving exactness of determining. The keen time arrangement anticipating framework that is utilized by sliding-window metaheuristic streamlining is a GUI that runs as an independent application. The framework make the financial exchange esteems expectation less difficult, including not many and not many calculations, than that utilizing the other strategy that were referenced. The first FA is enhanced with metaheuristic parts—turbulent maps, weight, versatile latency and Levy flight—to develop an advancement algo-rithm (MetaFA). The exhibition which is prevalent of the MetaFA was confirmed by benchmark capacities. In this manner, the MetaFA was embraced to tune the automatically the hyperparameters C and σ of the LSSVR. The streamlined expectation model was utilized with the sliding window to conjecture and assess stock cost. Default

setting of the framework, including prespecified estimations of parameters, spares the hour of clients. To assess the methodology which is proposed, it was applied to five datasets ,and three other stock datasets that have been utilized elsewhere. Factual measures were acquired when applied to development organization datasets. Specifically, the expectation of one day is 2597.TW Stock costs was better than any development organization stock costs, with a MAPE of 1.372%, a R of 0.990. Towards the finish of the investigation, the monetary presentation of the framework which was proposed was and analyzed, with empowering results. In this way, the proposed framework can be utilized as a choice taking device to figure stock costs for putting resources into present moment

The examination centers around the securities exchange .To sum up the utilization of the proposed framework, future work should utilize the proposed framework to evaluate stocks in other developing or develop markets, for example, Vietnam. At last, the advancement of an application which is online ought to be considered to improve the ease of use and ease of use of the master framework. The confinement of the framework is its computational speed, particularly as for approval of sliding window, in view of the intricacy of illuminating enormous science circle in the MATLAB program. The computational cost ascends with the quantity of approvals. Another shortcoming is the need of characterizing parcel of parameters of the framework however the default settings as gave. Besides, the framework didn't accomplish remarkable outcomes for long haul venture—a finding that will be found in future inquires about

2.1.4

<i>Title</i>	Incorporating a piecewise straight portrayal strategy and a neural system model for stock exchanging focuses expectation [4]
<i>Authors</i>	<p>Cha Pei:</p> <p>Fan Yuan:</p> <p>Chen-Hao Liu :</p> <p>.</p>
<i>Year of Publications</i>	02 December 2008
<i>Publishing Details</i>	IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)
<i>Summary</i>	<p>An ordinary measure of research was directed to contemplate the conduct of development of stock cost. In any case, the financial specialist is keen on making advantage or benefit by being given simple exchanging choices, for example, hold/purchase/sell from the framework rather choosing the stock value itself. Subsequently, an alternate strategy is made by applying PLR to shape various classifications of past information. As a discovering, turning or evolving focuses (trough or pinnacle) of the recorded or the stock information can be resolved and afterward be utilized as contribution to the BPN to prepare the association weight of the model. At that point, another arrangement of information is input that can actuate the model when a sell or purchase point is became acquainted with by the BPN. A savvy PLR model is then framed by joining the GA with the PLR. This improves the edge or the base estimation of PLR to additionally raise the benefit of model. The IPLR article is tried on different kinds of stocks, i.e., upturn, consistent, and downtrend. The tried outcomes show that the IPLR approach can make an adequate measure of benefit/advantage particularly on upswing and downtrend states as opposed to consistent state. By and large, the proposed framework is successful in its expectations in regards to the</p>

	<p>future exchanging purposes of a specific stock. Notwithstanding, there is one issue that is the value variety of the stock. It is seen that if the variety of cost of the present stock is estimated either in an upswing or a downtrend, at that point it is better that we train our BPN with a coordinating example, i.e., either in a comparative downtrend or upturn period.</p>
--	--

2.1.5

<i>Title</i>	An epic quick recurrence calculation and its application in stock record development prediction[5]
<i>Authors</i>	<p>Liming Zhang</p> <p>Pengyi Yu</p>
<i>Year Publications of</i>	11 May 2012
<i>Publishing Details</i>	IEEE Journal
<i>Summary</i>	<p>A very quick recurrence called - tallying IF is proposed. The recently characterized IF relieves the three necessities for sure fire recurrence. The rule is simple and straight forward. It tends to be utilized to make a straightforward wave, including IMFs got from an EMD calculation. Its theoretical fundamental and being basic and wide use make it to be of free intrigue. As the significant application is proposed - checking IF is then used to anticipate or on the other hand break down stock list utilizing a use of EMD disintegration. It is anticipated that the - checking IF techniques</p>

	may have significance in applications in money related information investigation.
--	---

2.1.6

<i>Title</i>	Neurofuzzy with half and half elements of instability and sign forecast [6]
<i>Authors</i>	D. Bekiros
<i>Year of Publications</i>	06 October 2011
<i>Publication details</i>	IEEE Transactions on Neural Networks
<i>Summary</i>	<p>Dependable anticipating strategies for budgetary applications are significant for financial specialists either to make benefit by exchanging or break against potential market dangers. In this paper the productivity of an exchanging plan depend on the usage of a neurofuzzy model is checked, so as to decide the heading of the market stock trade returns. Additionally, it is demonstrated that the combination of the appraisals of the restrictive unpredictability changes, as per the hypothesis of Bekaert and Wu, firmly improves the consistency of the neurofuzzy model, as it gives suitable data to a potential defining moment on the future exchanging day. The general return of the proposed unpredictability based neurofuzzy model including give and take(transaction) costs is reliably better than that of a Markov-exchanging model, a forward neural system just as a purchase and hold plan. The discoveries can be demonstrated by summoning either the "unpredictability criticism" hypothesis or the presence of portfolio protection plots in the business sectors of value and are likewise reliable with the view that instability reliance produces sign reliance.</p> <p>In this manner, an exchanging technique reliant on the proposed</p>

	neurofuzzy model may permit financial specialists to acquire a bigger number of profits than the latent portfolio the executives plan.
--	--

2.1.7

<i>Title</i>	Expectation of financial exchange execution by utilizing distinctive ML methods [7]
<i>Authors</i>	Karman Hasan
<i>Year of Publications</i>	04 May 2017
<i>Publishing Details</i>	2017 International Conference
<i>Summary</i>	<p>One choice can have enormous effect on a financial specialist's life in Stock Market. The financial exchange is an extremely perplexing framework and frequently a riddle, so it is , hard to dissect all the components affecting before settling on a choice. In this exploration, they have planned a securities exchange forecast model dependent on various variables. The model was worked to anticipate KSE-100 list execution. The market can be negative or positive with various qualities as anticipated by the model. The components included are value, loan cost, ware, outside trade, overall population feeling, variance of fuel anticipated qualities with assistance of chronicled or the past information of the market. The strategies utilized for expectation incorporate four distinct forms of Artificial Neural Network (ANN) including Radial Basis Function (RBF). All the strategies were contrasted with locate the best anticipating model. The outcomes indicated that MLP performed best and anticipated the market with precision of 77%. Each factor was concentrated freely to discover the relationship with advertise execution. The adjustment in Petrol costs demonstrated the most grounded relationship with advertise execution. The outcomes proposed that conduct of market can be anticipated utilizing AI methods.</p>

2.1.8

<i>Title</i>	Support Vector Machine used in Stock Market Prediction [8]
<i>Authors</i>	Zhen Hu Jie Zhu Ken Tse
<i>Year of Publications</i>	09 January 2014
<i>Publishing Details</i>	6th International Conference
<i>Summary</i>	A great deal of studies give proof of confronting noteworthy measure of difficulties in out-of test consistency tests because of vulnerability in model and flimsiness in parameter. As of late the presentation of certain methodologies that beat these issues are found. Bolster Vector Machine (SVM) is moderately another learning calculation having the attractive qualities of controlling the choice capacity, the utilization of the portion technique, and the sparsity of the arrangement. In this paper, they introduced an observational and hypothetical structure for applying the SVM methodology to anticipate the stocks. Right off the bat, some organization explicit and six macroeconomic elements that may impact the stock pattern are chosen for additional multivariate investigation. Besides, Support Vector Machine is utilized in finding the relationship of these components and foreseeing the performance. The result recommends that

	SVM is a force sponsor instrument in anticipating the monetary market for stock.
--	--

2.1.9

<i>Title</i>	Protections trade Prediction assessment by merging social and news end and conclusion [9]
<i>Authors</i>	Zhaoxia Wang Seng-Beng Ho Zhiping Lin
<i>Year of Publications</i>	11 February 2019
<i>Publishing Details</i>	Conference on Data Mining Workshops
<i>Summary</i>	The cost of stock is a decent marker for an organization and their numbers can be influenced by numerous components. Various occasions are influenced by open assessments and feelings in an unexpected way, which may influence the pattern of securities exchange costs. In light of reliance of different elements, the stock costs are not changeless, however are rather unique,. Because of its higher learning capacity for understanding the nonlinear time arrangement forecast issues, AI has been applied to the examination region. Learning-based techniques for stock value expectation are known and a great deal of achieved procedures have been utilized to improve the consequences of the learning based indicators. Notwithstanding, doing the effective securities exchange expectation is as yet an errand. News stories and online life information are additionally helpful and significant in money related expectation, yet right now nothing but bad strategy exists that can contemplate these web based life to give

better examination of the monetary market. This paper attempts to effectively foresee stock cost through thinking about the connection between the stock cost and the news. Contrasted and as of now introduced learning-based strategies, the adequacy of this new upgraded learning-based strategy is appeared by utilizing the genuine stock value informational collection with an improvement of execution as far as diminishing the Mean Square Error (MSE). The discoveries of this paper not just attempt to tell the benefits of the proposed technique, yet attempts to bring up to the right course for future work in this degree moreover.

2.1.10

<i>Title</i>	AI Techniques for Stock Price Prediction [10]
<i>Authors</i>	Sumeet Sarode Harsha G. Tolani Prateek Kak

Year of Publications	21 November 2019
Publishing Details	Conference on ICISS
Summary	<p>In recent time economies, there is a measurable effect of the stock or value advertise. Expectation of stock costs is extremely muddled, turbulent, and the nearness of a unique culture makes it a troublesome test. Social method of fund proposed that dynamic procedure of speculators to an extremely huge degree impacted by the notions and feelings because of a specific news. In this way, to help the choices of the financial specialists, we have a methodology joining two unique fields for investigation of stock trade. The framework consolidates value expectation dependent on past and constant information alongside news examination. LSTM (Long Short-Term Memory) is utilized for foreseeing. It requires the most recent exchanging data and examination quantifies as its info. For news investigation, just the related and live news is gathered from a major arrangement of business news. The arranged news is examined to anticipate notion of around organizations. The consequences of the two examinations are consolidated together to get an outcome which gives a suggestion for future ascents.</p>

Chapter-3

System Development

Third chapter includes system development. We have mentioned how the model and our project have been evolved over the time. We have drawn several flow charts for model extraction and system extraction for better understanding of the project

3.1 Approach using Support Vector Machine (SVM)

We are going to use Support vector machines (SVMs) for supervised learning methods as for categorization, reverting and outliers detection.

“Support Vector Machine” (SVM) is a directed AI calculation and we can utilize it for order and relapse issues yet we ordinarily use it for grouping issues. In SVM calculation, information things are plotted. Then, hyper-plane is identified by performing classification that differentiates the planes into two halves.

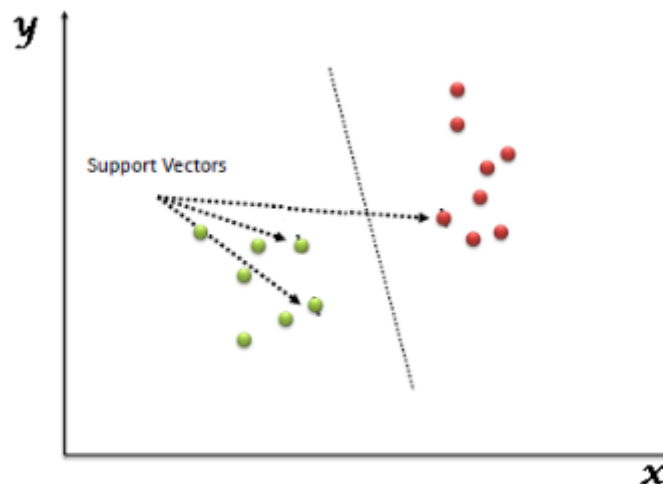


Fig3.1: SVM Diagram

Support Vectors are simply the co-ordinates of individual observation. Support Vector Machine is a frontier which best segregates the two classes (hyper-plane/line).

SVM has the following advantages:

- a. Effective in high dimensional spaces.
- b. It is helpful in places where amount of tests are less than measure of measurements
- c. Uses a subset of training points in the decision function (called support vectors), so it is also memory efficient.
- d. Versatile: SVM has the advantage of determining diverse piece capacities. We as a rule give normal bits however we can also specify custom kernels.

The disadvantages of support vector machines are:

- a. If the features are greater than the samples, avoid over fitting in choosing Kernel functions and regularization term is crucial.
- b. SVMs don't directly provide estimates of profitability, these are solved using an expensive five-fold cross-validation.

The support vector machines in scikit-learn support both dense and sparse sample vectors as input. Data must have been fit properly in order to fit predictions for sparse data. We have used C-ordered `numpy.ndarray(dense)` or `scipy.sparse.csr_matrix(sparse)` having `dtype=float=64` for optimal performance..

3.2 Classification and Regression

The picture beneath shows SVC, NuSVC and LinearSVC are classes that are fit for performing multi-class characterization on a dataset

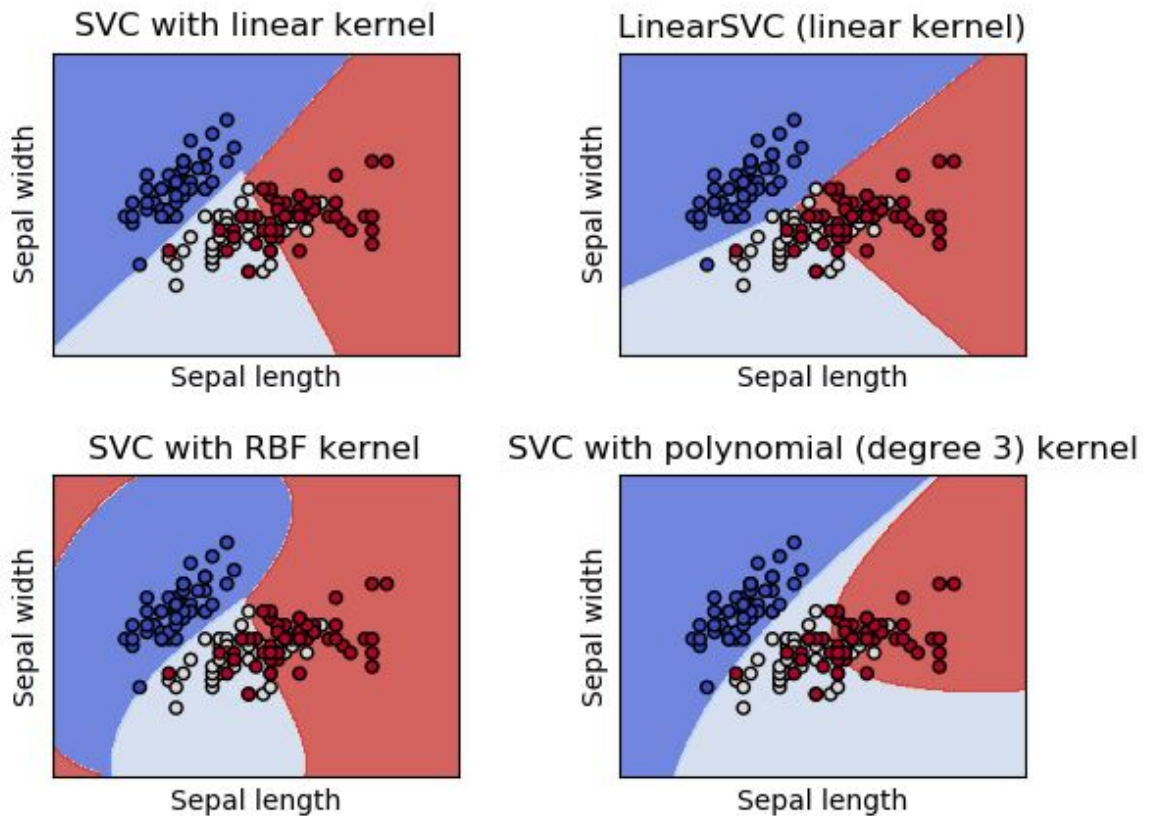


Fig 3.2: Classification of Data.....[a]

SVC and NuSVC are comparable strategies, however they have slight contrast in the arrangement of parameters and furthermore, they have diverse numerical plans. On the other note, LinearSVC is some another execution of Support Vector machines for the instance of a straight part. In LinearSVC, Kernel is thought to be direct and that is the reason it doesnot acknowledge catchphrase piece.

As other classifiers, **SVC**, **NuSVC** and **LinearSVC** take two arrays as input: an array X of size [n_samples, n_features] which will hold the training samples, and second array y of class labels (strings or integers), size [n_samples]:

```
>>from sklearnimportsvm
```

```
x= [[0, 0], [1, 1]]
```

```
y = [0, 1]
```

```
clf=svm.SVC(gamma='scale')
```

```
clf.fit(x, y)
```

```
SVC(C=1.0, cache_size=200, class_weight=None, coef=0.0,
```

```
decision_function_shape='ovr', degree=3, gamma='scale', kernel='rbf',
```

```
max_iter=-1, probability=False, random_state=None, shrinking=True, tol=0.001,
```

```
verbose=False)
```

```
clf.predict([[2., 2.]])
```

```
array([1])
```

get support vectors

`clf.support_vectors_array([[0., 0.], [1., 1.]])`

get files of help vectors

`clf.support_array([0, 1]...)`

get number of help vectors for each class

`clf.n_support_array([1, 1])`

3.2.1 Classification SVM Types

3.2.1.1 Classification SVM Type1

In this type of SVM, training involves error function minimization:

$$\frac{1}{2} w^T w + C \sum_{i=1}^N \xi_i \quad \text{eq. 1}$$

In limitations with:

$$y_i (w^T \phi(x_i) + b) \geq 1 - \xi_i \text{ and } \xi_i \geq 0, i = 1, \dots, N \quad \text{eq. 2}$$

where C is the breaking point consistent, w is the vector of coefficients, b is a predictable, and addresses boundaries for managing non particular data (inputs). The rundown names the N getting ready cases. Note that addresses the class names and xi addresses the self-sufficient elements. Data input is moved to incorporate space using part. More misstep is rebuffed if C regard is greater.. Thusly, C should be picked with care to keep up a key good ways from over fitting.

3.2.1.2 Classification SVM Type2

The Classification SVM Type 2 involves model minimizes the error function:

$$\frac{1}{2} w^T w - \nu \rho + \frac{1}{N} \sum_{i=1}^N \xi_i \quad \text{eq. 3}$$

subject to the constraints:

$$y_i (w^T \phi(x_i) + b) \geq \rho - \xi_i, \xi_i \geq 0, i = 1, \dots, N \text{ and } \rho \geq 0 \quad \text{eq. 4}$$

In a relapse SVM, subordinate variable and free factor reliance is evaluated.. It expect, as other relapse issues, the free and ward relationship is distinguished by deterministic capacity f(x) including a portion of the added substance commotion.

3.2.2 Regression SVM

3.2.2.1 Regression: SVM Type1

$$y = f(x) + \text{noise} \quad \text{eq.5}$$

We need to find a useful structure for $f(x)$ which will adequately foresee new cases that the SVM has not been given already. This can be cultivated by means of setting up the SVM model on a model set, i.e., getting ready set, a system that incorporates, like request the progressive improvement of a batch work.

3.2.2.2 Regression SVM Type2

The error function is given by:

$$\frac{1}{2} w^T w + C \sum_{i=1}^N \xi_i + C \sum_{i=1}^N \xi_i^* \quad \text{eq. 6}$$

which we minimize subject to:

$$\begin{aligned} w^T \phi(x_i) + b - y_i &\leq \varepsilon + \xi_i^* \\ y_i - w^T \phi(x_i) - b &\leq \varepsilon + \xi_i \\ \xi_i, \xi_i^* &\geq 0, i = 1, \dots, N \end{aligned} \quad \text{eq. 7}$$

Support Vector Machines models can use many number of kernels. These include polynomial, linear, radial basis function (RBF) and sigmoid.

3.3 Abstraction based extraction

Support Vector Machines has a basic idea dependent on recognizing of choice planes that characterize choice limits. A choice plane is one that isolates between various arrangement of articles with participations of various classes. The picture beneath delineates that. The picture, the items have a place either with class GREEN or RED. The limit that characterizes the different lines on the correct side of articles are GREEN and left side items are RED. Another article which is on the correct side is stamped, i.e., characterized, as GREEN and left one is delegated RED

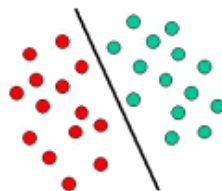


Fig 3.3(a): Abstraction

The above model is of a direct classifier, i.e., a classifier isolating two arrangements of articles as GREEN and RED. Most arrangement errands, in spite of the fact that, are not unreasonably basic, and frequently progressively complex structures are required so as to make an ideal partition, i.e., accurately grouping new articles (test information) based on the models that are accessible (train information). This circumstance is appeared in the picture below. When past schematic are being analyzed, plainly a full partition of the GREEN and RED articles would require a bend (which is more perplexing than a line). Order undertakings that depend on attracting isolating lines to recognize objects of various class participations are known as hyperplane classifiers. Support Vector Machines are structured in such an away to deal with such undertakings.

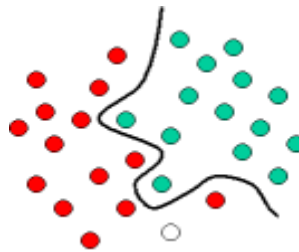


Fig 3.3(b): Abstraction

The picture underneath shows the fundamental structure behind Support Vector Machines. Here we can see the first items (left half of the schematic) mapped, that are revised, utilizing a lot of scientific capacities which are known as portions. The way toward reworking the items is known as mapping transformation. In the new setting, the mapped objects (right half of the outline) is straightly distinguishable along these lines, rather than building the unpredictable bend (left delineation), we need to do is to locate an ideal line which will isolate the GREEN and the RED items.

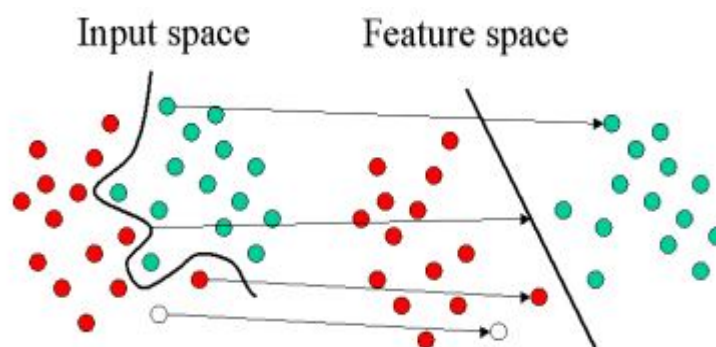


Fig 3.3(c): Abstraction

3.4 Parameters

There are two types of SVR: Linear and Multiple.

Tuning parameters esteem for calculations in Machine Learning emotionally improves the model execution. There are rundown of parameters accessible with SVR, which we will examine, so some significant parameters having higher effect on model execution, 'Part', 'Degree', 'Gamma', 'C'.

- a. Kernel: It is a similarity function and requires two inputs and spits out how similar they are. It helps in representing the infinite set of discrete function in a family of constant function.
- b. Gamma: It is used in RBF (Radial Basis Function) model to indicate variance. A small gamma means a Gaussian surface with large variance. Gamma controls the shape of peaks and height of pointed curves, higher the value of gamma, will try to exact fit the training datasets that is generalization error and cause overfitting problem.
- c. C: It is a penalty parameter for error term. To have the best fit some points in regression can always be ignored this is indicated by c or C means low bias and high variance as you penalise a lot misclassification. It also controls the trade-off between smooth decision boundary and classifying the training points correctly.
- d. Degree: Degree parameter specifies the degree of poly Kernel function. There is a trail and dealing with degree parameter. The more the degree parameter, more the accuracy but this also leads to more computational time and complexity.

3.5 Approach Using Linear Regression

Straight Regression being the underlying sort of relapse investigation to be thoroughly contemplated, and to be broadly utilized by and large applications. This is because of models or applications which relies straightly upon their obscure parameters are simpler to fit than models which are non-directly identified with their parameters and in light of the fact that the measurable properties of the subsequent estimators are simpler to decide.

On the off chance that the objective is expectation, estimating, or mistake decrease straight relapse may be utilized to fit a prescient model to a watched informational collection of estimations of the reaction and logical factors. In the wake of growing such a model, if extra estimations of the logical factors are gathered without a going with reaction esteem, the fitted model can be utilized to make a forecast of the reaction.

3.5.1. Proposed Algorithm

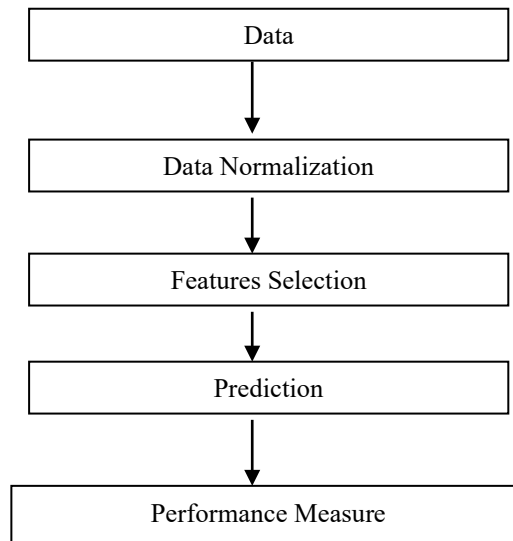


Fig 3.4: Linear Regression Algorithm

3.6 Approach using Decision Tree Analysis

A Decision Tree Analysis is a scientific model and is often used to make decisions in an organization. The graphic presentation is being shown by a tree type structure or building in which the issues can be checked in the form of flowchart, each with options or branches of alternating choices.

Decision Trees are very nice tools for helping to choose between several courses of actions. In Stock Prediction, the features are extracted from the daily stock market data, and then the related features are selected using decision tree. An approx set based classifier is used then to predict the next day's trend.

3.6.1. Terminologies used

a. Root Node: A root nodes means the whole sample, it is then divided into multiple sets which is made up of homogenous or similar variables.

b. Decision Node: A sub node that diverges or parts away into furthermore possibilities or chances is known as decision node.

c. Terminal Node: The final node showing the output o the outcome which can't be categorized further, is a leaf or terminal node basically where everything terminates.

d. Branch: it denotes the various alternatives or options available with decision tree making person.

e. Splitting: The division or separation of the available choice into the multiple sub nodes is what is known as splitting.

f. Pruning: Its just the opposite or vice versa of splitting, where the person making decision can eliminate or discard one or more sub-nodes from a particular decision node.

3.6.2 Steps in Decision Tree

Steps in Decision Tree Analysis

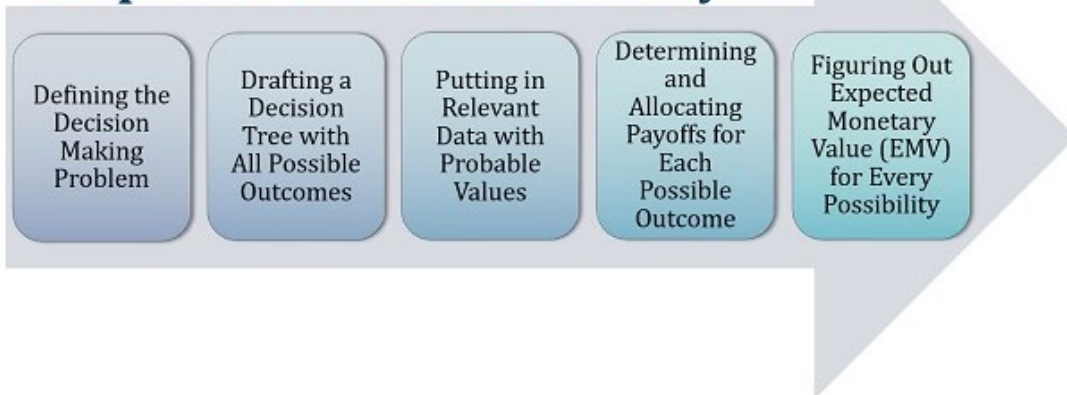


Fig 3.5: Steps of Decision Tree.....[b]

3.6.3 Representation

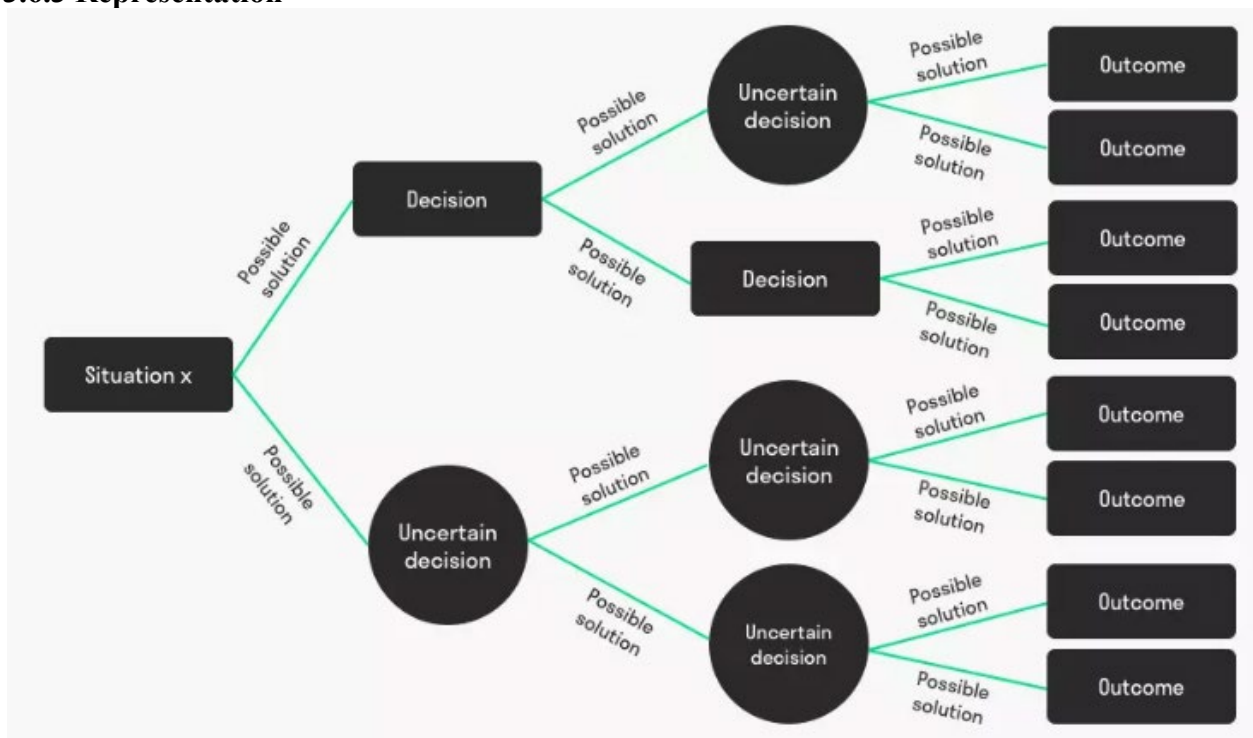


Fig 3.6:Representation

3.7 Proposed System for Extractive approach

This phase would involve supervised classification methods like Support Vector Machines, Neural Networks, Naive Bayes, Ensemble classifiers (like Adaboost, Random Forest Classifiers), etc.

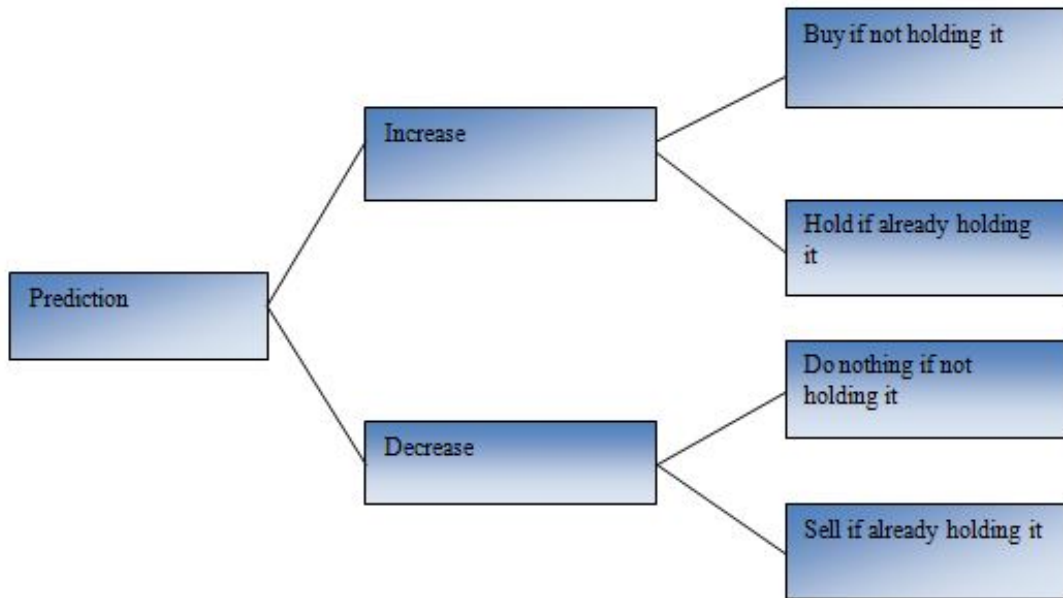


Fig 3.7: Proposed System

3.8 System architecture for Extractive approach

We are using SVM also known as Support Vector Machines in our project. SVM will make classification errors within training data in order to minimise overall error across test data.

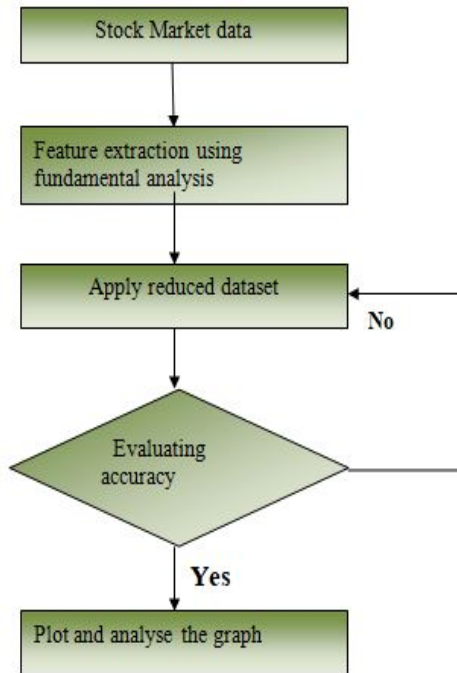


Fig. 3.8: System Architecture

3.9 Model Design

The proposed approach uses machine and deep learning concepts. The flow chart for this approach is as follows:

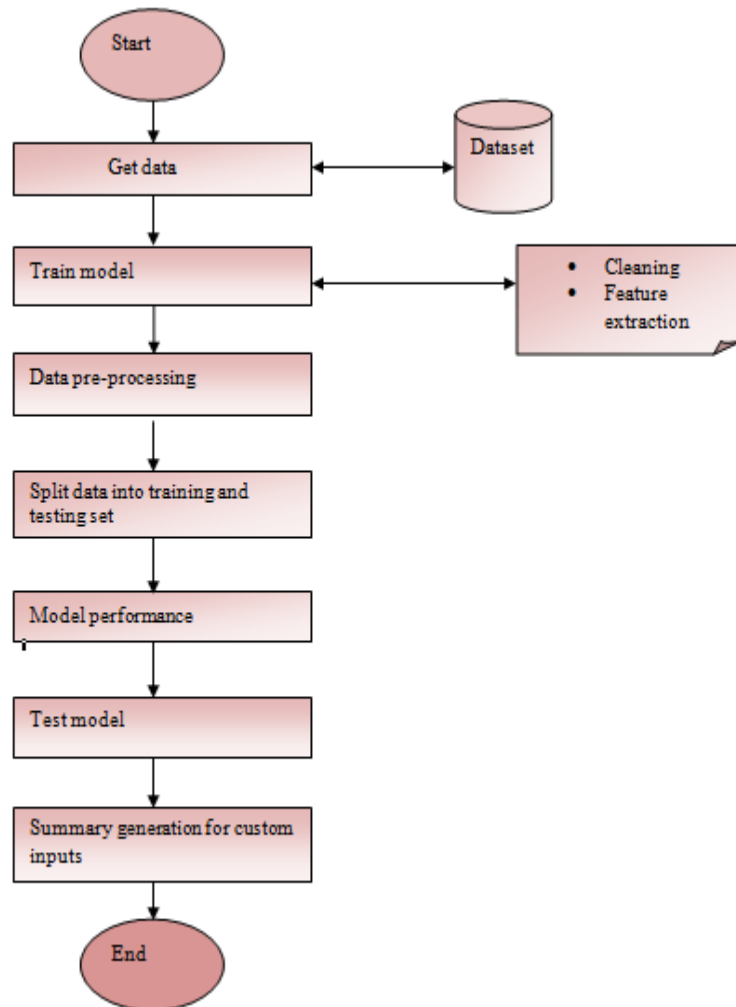


Fig 3.9: Model Design

Chapter-4

Performance Analysis

This chapter includes all the algorithms and mathematical formulas used in our project. Different steps used in applying algorithms. Result and result analysis and their accuracy have been scrutinized in this section.

4.1 Proposed solutions

a. Preprocessing and Cleaning

Removing the redundant data and recovering and the missing data and. This step basically involves creation of useful features form the existing ones.

b. Feature Extraction

In this step searching is done with the space of possible feature subsets. We then picked up the subset which is optimal or near-optimal with respect to some objective function. Overfitting and underfitting the dataset is major problem and hence, this is done to avoid the same.

c. Data Normalization

Information is should have been standardized for better exactness by guaranteeing that all highlights are not given over the top/low weight age.

d. Analysis of various supervised learning methods

d.1. Classification Methods

Support Vector Machines, Neural Networks, Naïve Bayes, Ensemble classifiers (like Vector Machines, Neural Networks, Naive Bayes Adaboost, Random Forest Classifiers) are all part of this phase which are part of supervised learning methods.

d.2 Regression Methods

These models are utilized for intrigued stocks to get the normal numerical value. This stage would include administered relapses methods like Linear Regressions, Support Vector Regressions, Usage of Kernel Methods, etc.

e. Social Media Sentiment Analysis

Analysing the situation of the current market from the facebook and twitter or form latest news headlines in order to gain insights into the future of stock prices.

f. Analysis of Different Models

Comparison between the various methods and models implemented over the datasets.

4.2 Analysis

- a. Analysis of stocks will be helpful for new merchants to exchange securities exchange depended on the different variables considered by the product application.
- b. Our programming will figure the sensex dependent on organization's stock worth. There are a great deal of components on which stock estimation of organization depends. Some of them are:

b.1 Demand and Supply: Supply of company's share is a major reason for change in price of stocks. Increased demand and decreased supply leads to increase in value and vice versa.

b.2 Corporate results: This will be in with respect to the benefits or progress of the organization over some stretch of time of not many months.

b.3 Fame: Main power to share buyer. Popularity of a company effect the ones buying.

4.2.1 Prominent features based analysis

a. Analyzing stock data.

We need to give dataset of a company, which will include its opening and closing price of monthly sales or profit.

b. Analyzing the factors.

We have to get the data in the same period for the following factors.

b.1 Demand and Supply: by the previous data entered.

b.2. Corporate results: Companies declare their results and profit at the last of each Quarter.

b.3. Popularity: Analysing the views about the company.

4.2.2 Prediction analysis

- a. Technical analysis
- b. Fundamental analysis

Specialized examination is a momentary system while basic investigation is long haul methodology. By evaluating on inborn qualities principal investigation permit us to progress in the direction of the drawn out estimation of the organization.

4.3 Performance Measures

- a. **R² Score (R-squared):** It is a statistical measure of how near or close or in the proximity of the data is to the fitted regression. 0% means or indicates that the data or model explains no variability of the response around its mean, or it simply means no difference or variation is there.
- b. **RMSE (Root Mean Squared Error):** It is the forecast blunder or the standard deviation of the residuals (these are the proportion of how long away or far the relapse line information focuses were). RMSE is a proportion of the distance away or spread out these residuals are, or it will going to reveal to you that how focused or close enough is the information close to the line or way of best fit. The little is the RMSE, the better is the model.
- c. **MSE (Mean Squared Error):** It is basically the average of differences square between the predicted and actual values, but it pressurizes more on importance of large errors. Also it should be taken care that less MSE is better for model or the balance between over and under fit.
- d. **MAE (Mean Absolute Error):** It means the results of measuring the difference between two variables which are continuous. It is better than RMSE if big mistakes or errors are undesirable.

4.4 Training Dataset

This dataset includes the dates, open, high, low, last, closing prices, total trading amount and turnover.

Date	Open	High	Low	Last	Close	Total Trad	Turnover (Lacs)
10-08-2018	208	222.25	206.85	216	215.15	4642146	10062.83
10-05-2018	217	218.6	205.9	210.25	209.2	3519515	7407.06
10-04-2018	223.5	227.8	216.15	217.25	218.2	1728786	3815.79
10-03-2018	230	237.5	225.75	226.45	227.6	1708590	3960.27
10-01-2018	234.55	234.6	221.05	230.3	230.9	1534749	3486.05
9/28/2018	234.05	235.95	230.2	233.5	233.75	3069914	7162.35
9/27/2018	234.55	236.8	231.1	233.8	233.25	5082859	11859.95
9/26/2018	240	240	232.5	235	234.25	2240909	5248.6
9/25/2018	233.3	236.75	232	236.25	236.1	2349368	5503.9
9/24/2018	233.55	239.2	230.75	234	233.3	3423509	7999.55
9/21/2018	235	237	227.95	233.75	234.6	5395319	12589.59
9/19/2018	235.95	237.2	233.45	234.6	234.9	1362058	3202.78
9/18/2018	237.9	239.25	233.5	235.5	235.05	2614794	6163.7
9/17/2018	233.15	238	230.25	236.4	236.6	3170894	7445.41
9/14/2018	223.45	236.7	223.3	234	233.95	6377909	14784.5
09-12-2018	216.35	223.7	212.65	221.65	222.65	4570939	10002.01
09-11-2018	222.5	225.4	214.85	216.35	216	3508990	7735.81
09-10-2018	222.5	235.15	220.65	221.05	222	7514106	17130.29
09-07-2018	221	224.5	219.1	223.15	222.95	1232507	2742.84
09-06-2018	224	225	218.2	220.95	221.05	1738824	3856.72
09-05-2018	222	224.6	215.2	222.1	222.4	3023097	6674.93
09-04-2018	238.2	238.2	222.6	223.45	223.7	3554859	8163.82
09-03-2018	236	243.55	235.05	236.85	236.7	5242852	12538.39
8/31/2018	237	239.75	232.95	234.65	234.3	3353833	7913.21

8/30/2018	235.35	237.3	232.1	237.3	236	1921327	4516.57
8/29/2018	233.85	237.7	232.7	234.2	234.55	1394661	3280.33
8/28/2018	237	239.3	231.3	232.9	233.35	2374782	5571.77
8/27/2018	231.8	239.35	231.05	236.8	237.05	1990020	4689.94
8/24/2018	234.5	237.2	230.2	231.5	231	1838417	4289.35
8/23/2018	240.3	240.6	233.1	235.5	235.45	1553953	3662.36
8/21/2018	246.9	246.9	239.25	240.9	240.55	3272005	7941.4
8/20/2018	244	247	243	244.7	245.15	1690225	4141.83
8/17/2018	240.8	244.5	239.2	242.7	243	2838238	6885.52
8/16/2018	236.05	242	235.95	240.35	239.35	2551480	6106.81
8/14/2018	235	238.5	235	237.4	237.55	1885288	4459.96
8/13/2018	233	236.45	232.25	235.2	234.55	1948583	4573.57
08-10-2018	237.3	237.95	231.1	233.65	233.55	2035594	4757.48
08-09-2018	236.65	239.85	235.3	237.25	237.3	1127248	2676.63
08-08-2018	237.25	240.5	235.05	236.35	236.35	1807313	4305.39
08-07-2018	241	241.55	235.3	237.5	237.6	1963538	4677.58
08-06-2018	235.15	240.45	234.15	240	239.5	2560616	6080.87
08-03-2018	236	239	232	235.2	234.65	3357945	7893.32
08-02-2018	232.5	238.05	230.4	235	235.45	5700851	13328.52
08-01-2018	248.7	254.95	234.35	235.1	235.65	13272609	32325.56
7/31/2018	243.4	249.65	241.5	247.1	246.9	4422342	10917.34
7/30/2018	243.7	247	240.9	243.65	242.2	1867720	4544.83
7/27/2018	241	243.95	239.15	242.5	242.25	1581985	3821.97
7/26/2018	235.9	242.95	234.5	239.5	239.1	3996921	9565.1
7/25/2018	248.5	248.5	235.8	236.05	236.9	2310430	5596.44

1184	12/23/2013	154.5	156.5	153.35	154.1	154.7	3101414	4817.62
1185	12/20/2013	147.2	153.85	146.6	152.85	153.1	5456617	8298.94
1186	12/19/2013	148.55	149.4	145.9	147.3	147.5	1833784	2704.13
1187	12/18/2013	145.3	148.75	145.25	147.85	148.2	1672571	2462.36
1188	12/17/2013	144.1	147.1	144.1	144.5	144.75	1678569	2443.6
1189	12/16/2013	145.25	145.5	142.2	143.45	143.45	1279705	1834.35
1190	12/13/2013	146	147.5	145	145.3	145.15	1102791	1608.29
1191	12-12-2013	146.8	148.45	146.2	146.55	146.75	1185509	1745.97
1192	12-11-2013	147.95	147.95	146.25	147.2	147	1285575	1889.48
1193	12-10-2013	146.75	149	146.7	147.6	147.45	1821331	2689.55
1194	12-09-2013	149.1	149.1	145.3	146.65	146.7	2508890	3674.98
1195	12-06-2013	146.8	147.75	145.8	146.15	146.2	1385102	2029.35
1196	12-05-2013	147.6	148.7	146.05	146.6	146.65	1565641	2304.46
1197	12-04-2013	146.5	148.4	145.1	145.9	145.55	1547460	2267.72
1198	12-03-2013	149	149.3	146.25	146.7	146.8	2227417	3286.14
1199	12-02-2013	149.2	150.9	149	149.25	149.3	2063513	3091.19
1200	11/29/2013	148	150.1	147.55	148.75	148.9	2020588	3005.77
1201	11/28/2013	145	148.35	145	148.2	147.65	3615167	5318.02
1202	11/27/2013	146.3	147.6	144	144.75	145.05	1855005	2699.14
1203	11/26/2013	142	146.8	141.6	145.8	145.5	2505101	3641.28
1204	11/25/2013	142.55	143.75	140.8	142.5	142.4	2305892	3285.91
1205	11/22/2013	144.35	145.2	141.1	142.35	141.8	2315264	3306.1
1206	11/21/2013	145.1	146.8	143.1	144.1	143.65	2997918	4355.49
1207	11/20/2013	146.25	147.8	144.75	145.8	145.35	2756146	4034.8
1208	11/19/2013	149	149.2	144.05	145	144.55	3835593	5596.79

1209	11/18/2013	142.5	148.35	142.5	147.5	147.7	5738832	8407.83
1210	11/14/2013	146.05	146.9	142.8	144.4	143.95	4246087	6161.68
1211	11/13/2013	155	155	141.1	144.9	144.3	11917625	17299.48
1212	11-12-2013	157.1	159.65	153.1	155.25	154.55	1842456	2891.59
1213	11-11-2013	160.4	161.25	155.1	156.75	156.55	1954941	3059.76
1214	11-08-2013	160.2	163	157.8	160.2	160.1	1175152	1883.24
1215	11-07-2013	164.4	167.9	159.05	161.5	160.35	2358582	3874.98
1216	11-06-2013	167.7	168.95	163.1	163.7	163.55	2080746	3437.64
1217	11-05-2013	168.5	169.8	165.9	167.8	167.6	1762917	2960.86
1218	11-03-2013	168	170	167.25	169.5	169.5	512878	868.66
1219	11-01-2013	164	169.9	161.55	168.3	167.7	3726470	6227.17
1220	10/31/2013	165	167.1	163.1	164.1	164	3033400	4991.85
1221	10/30/2013	163	166	162.65	165.3	165	1918758	3158.04
1222	10/29/2013	163.3	164.2	160.25	162.25	162.4	2067392	3350.14
1223	10/28/2013	164	166.25	162.3	163.1	163.25	1058274	1733.67
1224	10/25/2013	166.2	168.4	163.05	163.45	163.85	4412267	7337.93
1225	10/24/2013	159	166.4	158	165.75	165.45	4943284	8096.77
1226	10/23/2013	162.1	162.6	157	158.9	158.75	1228667	1950.57
1227	10/22/2013	160.4	162.8	159.9	162.25	161.85	1598301	2583.68
1228	10/21/2013	164.5	165.35	159	159.8	159.6	3257249	5252.89
1229	10/18/2013	163.2	165	162.25	164	164.2	2540836	4163.49
1230	10/17/2013	159.1	162.9	158.25	161.75	162	2724697	4381.6
1231	10/15/2013	160	160.2	155.35	157.1	158.05	1145582	1805.49
1232	10/14/2013	160.85	161.45	157.7	159.3	159.45	1281419	2039.09
1233	10-11-2013	161.15	163.45	159	159.8	160.05	1880046	3030.76

4.5 Testing Snippet

This will be dataset that we will test on.

This dataset includes the date, open, high, low, close, last, close or Adjust close and volume of the company i.e. total trade and turnover.

1	Date	Open	High	Low	Last	Close	Total Trad	Turnover (Lacs)
2	10-08-2018	208	222.25	206.85	216	215.15	4642146	10062.83
3	10-05-2018	217	218.6	205.9	210.25	209.2	3519515	7407.06
4	10-04-2018	223.5	227.8	216.15	217.25	218.2	1728786	3815.79
5	10-03-2018	230	237.5	225.75	226.45	227.6	1708590	3960.27
6	10-01-2018	234.55	234.6	221.05	230.3	230.9	1534749	3486.05

- a. **Date:** It's the date in the format MM-DD-YYYY, on which the trading is done, or prediction is taken place.
- b. **Open:** The worth every single offer has taken when the Stock trades opens up. It offers a decent hint or sign of where stock will go during the entire time it showcase is open. As the Stock trade can be connected with a closeout showcase so purchasers and venders make manages the most elevated bidder, so the opening and earlier day's end value need not to be same.
- c. **High/Low Price:** These are to be taken a day before and gives the sign of how much the share move during a day usually and how will it implicates the closing price, it basically shows the basic cyclic movement of a share.
- d. **Last Price:** It is used to tell the most recently which is reported trading price for the future contract.
- e. **Close:** It is the stock shutting cost of the specific date or day of exchanging that has been changed to incorporate any circulations and activities that should be joined that occurred whenever earlier or before to the following day open.
- f. **Total Trad:** the number of shares that are being traded in an entire market during given period of time.
- g. **Turnover:** Its measure of stock liquidity calculated as no of shares traded over a period per average number of shares outstanding for that period.

4.6 Custom Input

The necessary python files/dependencies are imported.

```
import pandas as pd
import numpy as np
from sklearn.svm import SVR
import matplotlib.pyplot as plt
```

Dependencies installed are as follows:

- a. Requests: Https request are handled by this library.
- b. Pandas as pd: for handling csv files and dataframes.
- c. Re: it is used for string operations

- d. Operator: used for numerical operations.
- e. Sys: used for system calls
- f. Urllib: used to put or get http url requests
- g. OS: used for instructing operating systems to perform functions on file system
- h. Csv:used to handle csv files, passing reading and writing in csv formats
- i. Numpy as np: used for scientific calculations
- j. From sklearn.svm import svr: used for machine learning
- k. Matplotlib.pyplot as plt: used for visualization and plotting the graphs

We have a function first to get the previous stock price data of the company

```
import pandas as pd
import numpy as np
#read Dataset from the storage
data = pd.read_csv("Stock.csv")

New_data = data

print(data.head())
```

Once we get the data, we load it in a CSV file for future processing

```
from datetime import datetime
solve_date = pd.to_datetime(data["Date"])
datess = solve_date.dt.year

New_data["Date"] = datess #Here we overwrite the daywise data into year wise

#Now here we have one categorical feature which is year so ML not handle this ,we convert it into normal form using encoding
category = pd.get_dummies(New_data.Date)

#Now we label encode the data and then one hot encode it
from sklearn.preprocessing import LabelEncoder,OneHotEncoder

l = LabelEncoder()
New_data["Date"] =l.fit_transform(New_data["Date"]) #now we overwrite years in new_data by label encoding

onehtencdr = OneHotEncoder(categorical_features=[0])
ds =onehtencdr.fit_transform(New_data).toarray()
```

Now, we take from dataset x and y value

```
dataset = pd.DataFrame(ds)
x = dataset.values[:, :-1]
y = dataset[12]
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.22)
```

Input for Linear Regression,

```

from sklearn.linear_model import LinearRegression

algo = LinearRegression()
model = algo.fit(x_train,y_train)

pred = model.predict(x_test)
pred = pd.Series(pred)

from sklearn import metrics
from sklearn.metrics import r2_score
print("(MAE):",metrics.mean_absolute_error(y_test,pred))
print("(MSE):",metrics.mean_squared_error(y_test,pred))
print("(RMSE):",np.sqrt(metrics.mean_squared_error(y_test,pred)))
print("the value of r2_score==",r2_score(y_test,pred))
print("*****Using Linear Reg.*****")
import matplotlib.pyplot as plt

plt.xlabel("Predicted Y")
plt.ylabel("Actual Y")
plt.scatter(y_test,pred)
plt.show()

z = list(range(1,101))

plt.scatter(z,pred[:100],s = 15)
plt.plot(z,pred[:100],label="pred")
plt.scatter(z,y_test[:100],s =15)
plt.plot(z,y_test[:100],label = "ytest")
plt.xlabel("Training Examples (1-100)")
plt.ylabel("Stock Price Values")
plt.legend()
plt.show()

plt.plot(y_test,color="red", label = "ytest")
plt.plot(pred,color="orange", label = "pred")
plt.xlabel("Values")
plt.ylabel("profit")
plt.legend()
plt.show()

plt.pie([y_test.mean(),pred.mean()],labels= ["mean of y_test = {}".format(round(y_test.mean(),2)), "mean of predicted y = {}".format(round(pred.mean(),2))],shadow=True)
plt.show()

```

Input for Support Vector Regression,

```
print("*****SVM*****")
from sklearn.svm import SVR

algo = SVR(kernel="rbf")
model = algo.fit(x_train,y_train)

pred = model.predict(x_test)
pred = pd.Series(pred)

from sklearn import metrics
from sklearn.metrics import r2_score
print("(MAE):",metrics.mean_absolute_error(y_test,pred))
print("(MSE):",metrics.mean_squared_error(y_test,pred))
print("(RMSE):",np.sqrt(metrics.mean_squared_error(y_test,pred)))
print("the value of r2_score===",r2_score(y_test,pred))

plt.scatter(y_test,pred)
plt.xlabel("Predicted y")
plt.ylabel("Actual y")
plt.show()

z = list(range(1,101))
plt.scatter(z,pred[:100],s = 15)
plt.plot(z,pred[:100],label="pred")
plt.scatter(z,y_test[:100],s =15)
plt.plot(z,y_test[:100],label = "ytest")
plt.xlabel("Training Examples (1-100)")
plt.ylabel("Stock Price values")
plt.legend()
plt.show()

plt.plot(y_test,color="red", label = "ytest")
plt.plot(pred,color="orange", label = "pred")
plt.xlabel("Values")
plt.ylabel("profit")
plt.legend()

plt.show()

plt.pie([y_test.mean(),pred.mean()], shadow=True,labels= ["mean of y_test = {}".format(round(y_test.mean(),2)), "mean of predicted y = {}".format(round(pred.mean(),2))])
plt.show()
```

Input for Decision Tree,

```
print("*****Decision Tree*****")

from sklearn.tree import DecisionTreeRegressor

algo = DecisionTreeRegressor()
model = algo.fit(x_train,y_train)

pred = model.predict(x_test)
pred = pd.Series(pred)

from sklearn import metrics
from sklearn.metrics import r2_score
print("(MAE):",metrics.mean_absolute_error(y_test,pred))
print("(MSE):",metrics.mean_squared_error(y_test,pred))
print("(RMSE):",np.sqrt(metrics.mean_squared_error(y_test,pred)))
print("the value of r2_score===",r2_score(y_test,pred))
plt.scatter(y_test,pred)
plt.xlabel("Predicted y")
plt.ylabel("Actual y")
plt.show()

z = list(range(1,101))

plt.scatter(z,pred[:100],s = 15)
plt.plot(z,pred[:100],label="pred")
plt.scatter(z,y_test[:100],s =15)
plt.plot(z,y_test[:100],label = "ytest")
plt.xlabel("Training Examples (1-100)")
plt.ylabel("Stock Price values")
plt.legend()
plt.show()

plt.plot(y_test,color="red", label = "ytest")
plt.plot(pred,color="orange", label = "pred")
plt.xlabel("Values")
plt.ylabel("profit")
plt.legend()
plt.show()

plt.pie([y_test.mean(),pred.mean()], shadow=True, labels= ["mean of y_test = {}".format(round(y_test.mean(),2)), "mean of predicted y = {}".format(round(pred.mean(),2))])
plt.show()
```

4.7 Output/Results

Initial Data fetched from Dataset or the csv. File

	Date	Open	High	Low	Last	Close
Total Trade Quantity \						
0	10/8/2018	208.00	222.25	206.85	216.00	215.15
4642146						
1	10/5/2018	217.00	218.60	205.90	210.25	209.20
3519515						
2	10/4/2018	223.50	227.80	216.15	217.25	218.20
1728786						
3	10/3/2018	230.00	237.50	225.75	226.45	227.60
1708590						
4	10/1/2018	234.55	234.60	221.05	230.30	230.90
1534749						

	Turnover (Lacs)
0	10062.83
1	7407.06
2	3815.79
3	3960.27
4	3486.05

Fig 4.1: Initial data

4.7.1 Linear Regression

Error terms for Linear Regression:

```
(MAE): 640.5670380089828
(MSE): 853132.6337238364
(RMSE): 923.6517924650157
the value of r2_score=== 0.9643738332484679
```

Fig 4.2:Error Term for LR

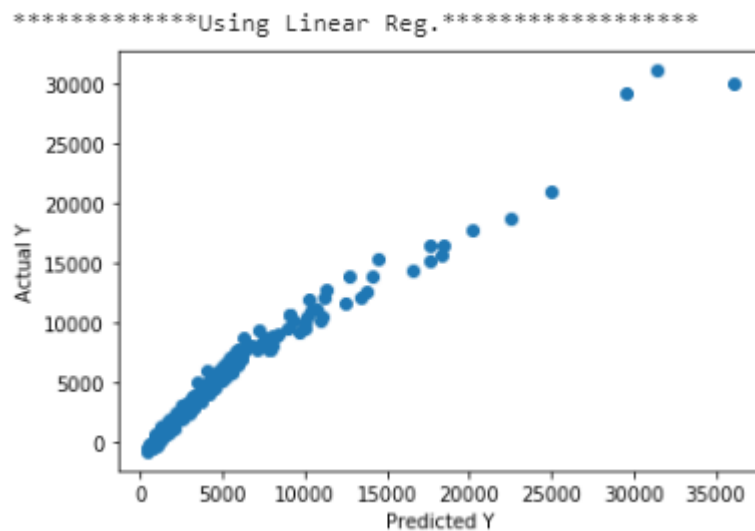


Fig 4.3: Graph between Actual and Predicted Y for Linear Regression

Graph between Stock Price Values v/s Training Examples (1-100) or between predicted and actual values for Linear Regression

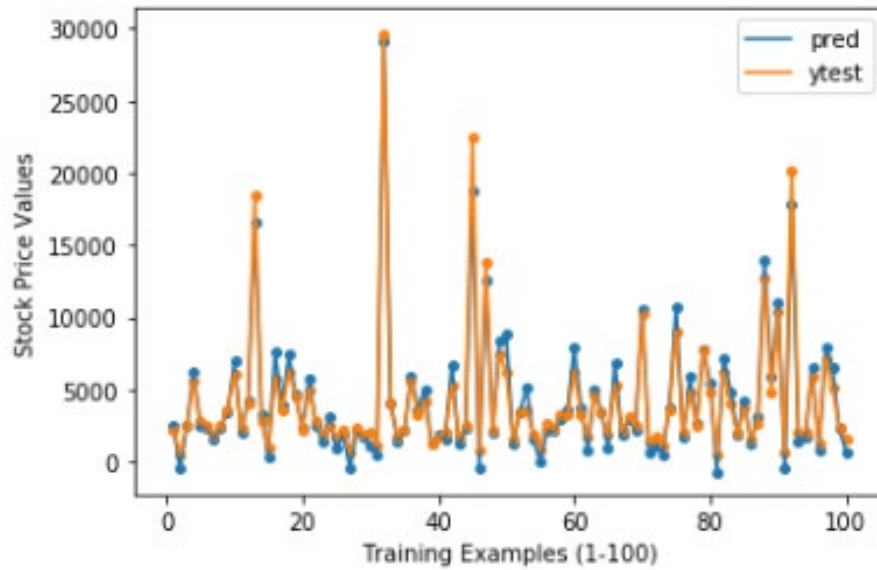


Fig 4.4: Graph between Stock Price Value v/s Training Examples(1-100) or between Predicted and Actual values for Linear Regression

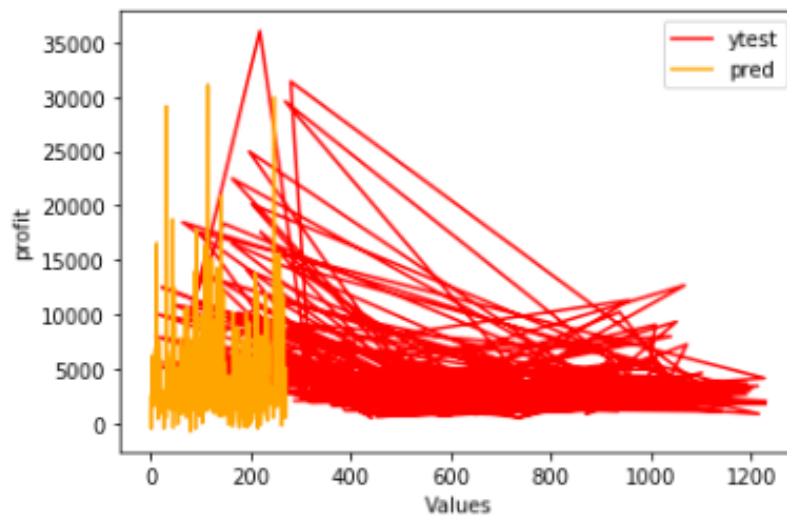
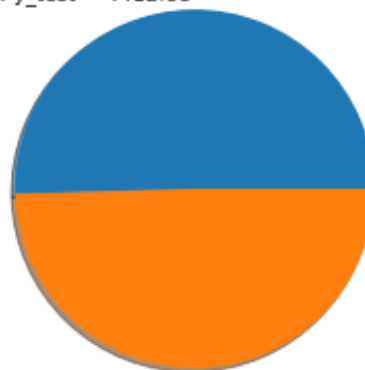


Fig 4.5: Graph of profit v/s values for Linear Regression

mean of y_{test} = 4412.88



mean of predicted y = 4336.03

Fig 4.6: Pie chart representation of actual and predicted for Linear Regression

4.7.2 Support Vector Machine

Error terms for SVM:

```
*****SVM*****  
(MAE): 2717.7608088235293  
(MSE): 25522384.345533088  
(RMSE): 5051.968363473102  
the value of r2_score=== -0.0657952640046211
```

Fig 4.7: Error Term for SVM

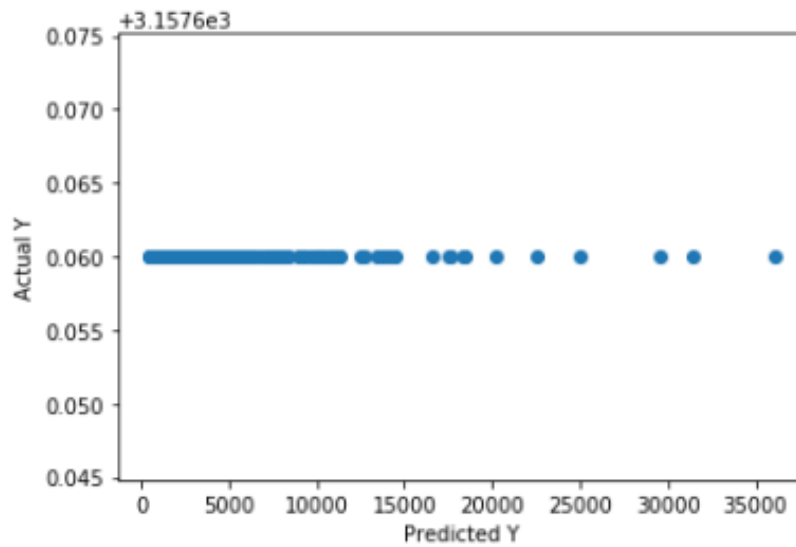


Fig 4.8: Graph between Actual and Predicted Y for SVM

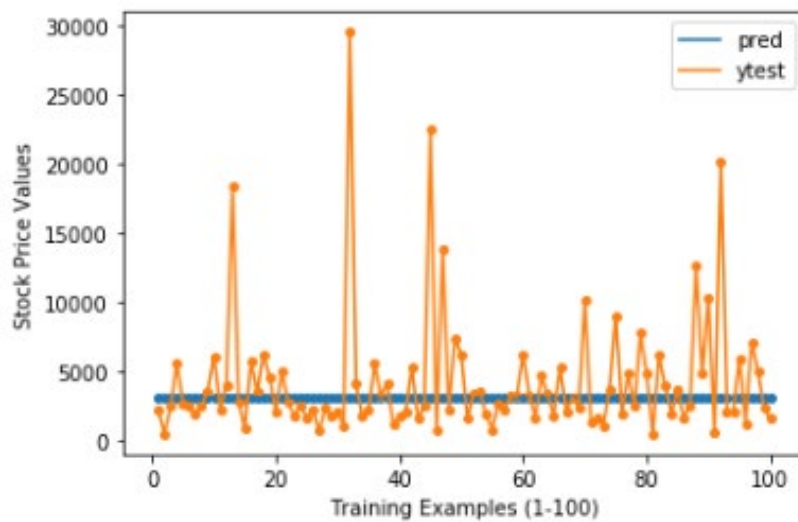


Fig 4.9: Graph between Stock Price Value v/s Training Examples(1-100) or between Predicted and Actual values for SVM

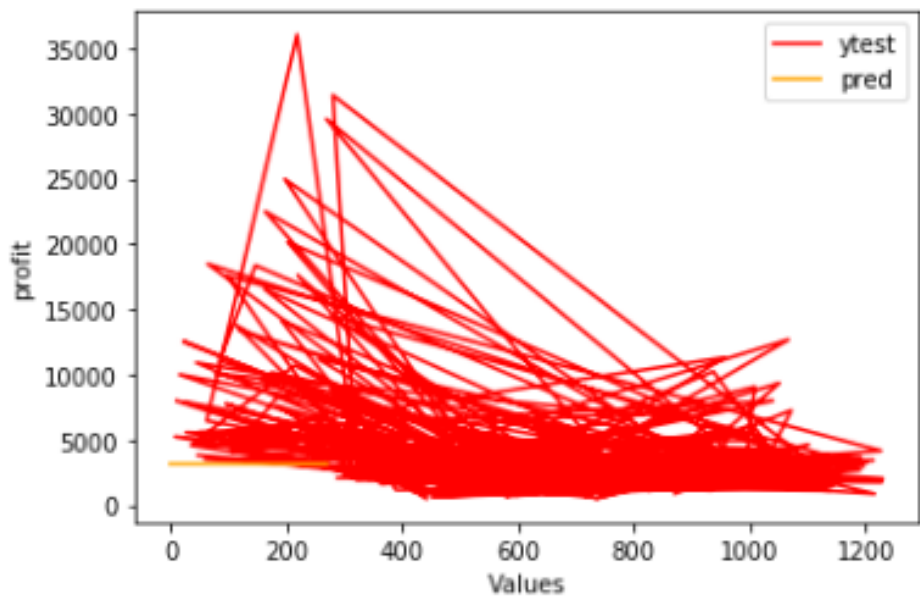


Fig 4.10: Graph of Profit v/s Values for SVM

Pie chart representation of actual and predicted for SVM

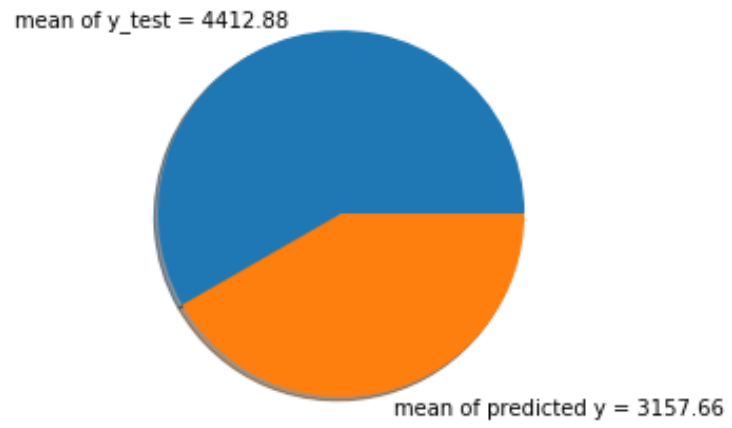


Fig 4.11: Pie chart representation of actual and predicted for SVM

4.7.3 Decision Tree

```

*****Decision Tree*****
(MAE): 243.6329411764706
(MSE): 694005.503933088
(RMSE): 833.0699273969071
the value of r2_score=== 0.9710188605707413

```

Fig 4.12: Error Term for DT

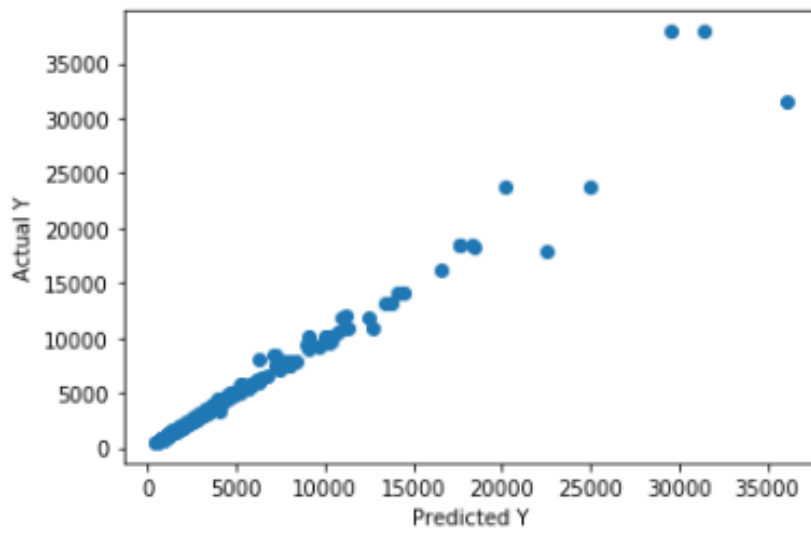


Fig 4.13: Graph between Actual and Predicted Y for Decision Tree

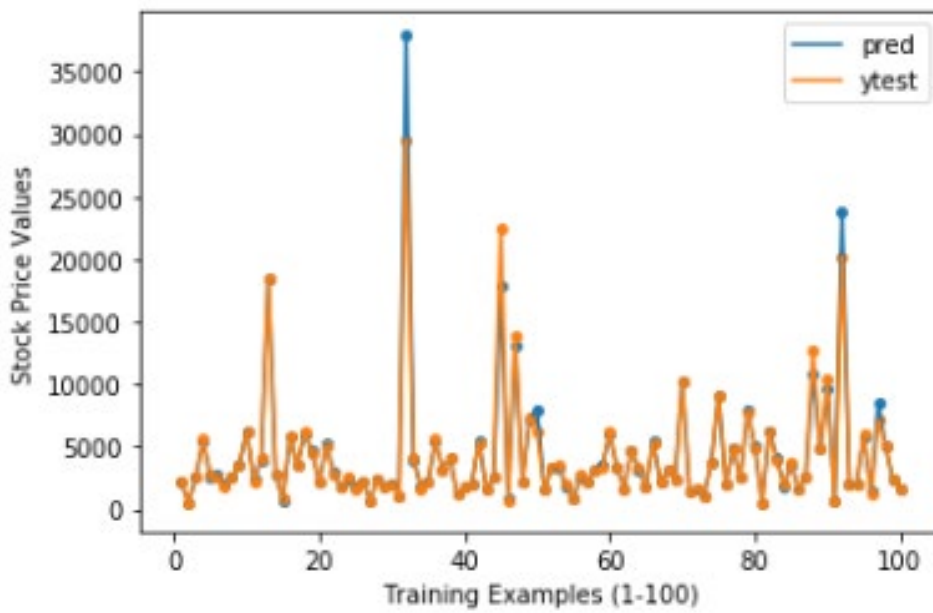


Fig 4.14: Graph between Stock Price Values v/s Training Examples(1-100) or between predicted and Actual values for Decision Tree

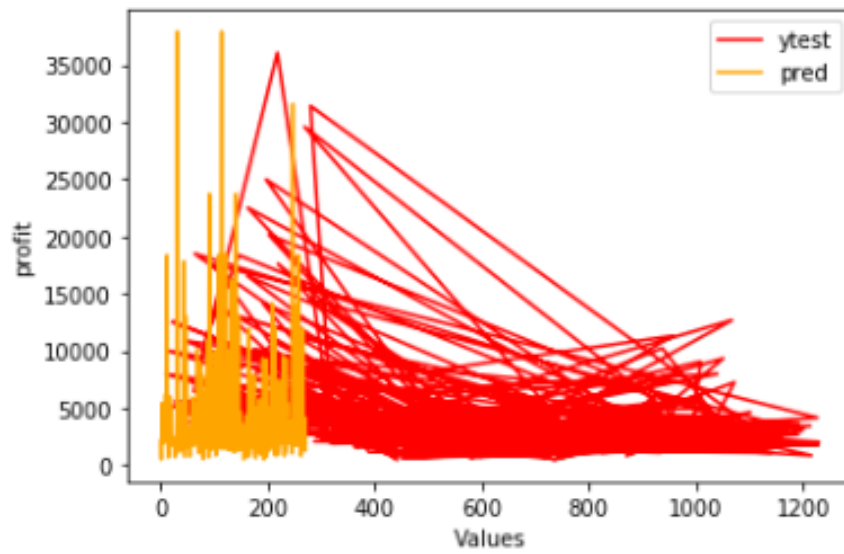


Fig 4,15: Graph of Profit v/s Values for Decision Tree

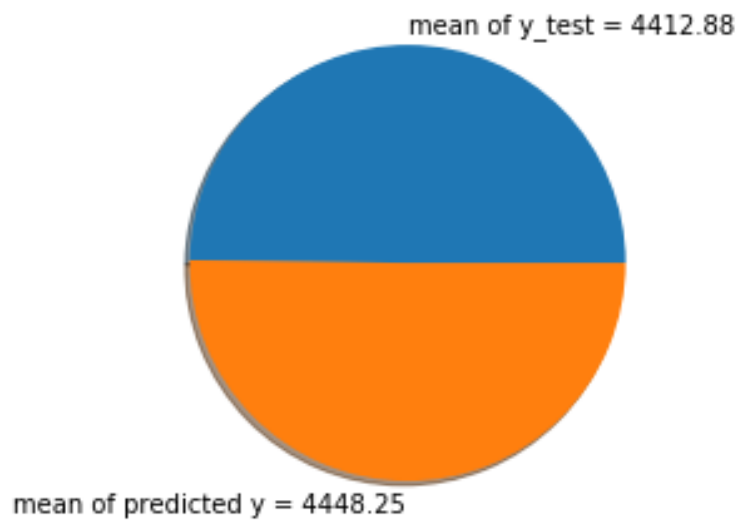


Fig 4.16: Pie Chart representation of Actual v/s Predicted for Decision Tree

Chapter 5

Conclusion

Internet has a growing rate and the rate with which the data is being generating , it has become almost impossible for us to handle and take care of such data. Such an enormous amount of information is processing nowadays that it becomes difficult for us to study their behavior or to conclude anything from them, thus making it so hard to summarize it. Then comes Machine Learning algorithms that helps us in understanding such datasets. Technical and fundamental analysis have showed a little work in the experiments carried out. Machine learning algorithm was applied to various data sources of different companies. Report highlights that stock market is prone to differences. Report also concludes that predicting stock prices is extremely tough job.

The main objective of this system is providing ways to heal the stock market. Our task is such that it can't be used for official model because of its limitedness. We have reached to a certain degree of accuracy by incorporating the limited number of parameters. Since stock market is highly fluctuating so to predict everything with great or large accuracy can't be taken into account. So our model that we have created has only depends on the selected number of parameters and their relationship with the share price.

By this basic learning of Extractive and Abstractive Method and tried to implement the initial one. We have successfully taken a dataset of different companies and performed data cleaning and normalization. Then we have split the dataset into testing and training in which testing dataset is almost 10%. After that, we have created Linear Regression, support regression model and Decision Tree algorithm to trained our model on dates and prices.

Predicting the stock market forecast is always challenging and a tedious job, specially a challenging work for business analysts. We have calculated our prediction with an overall accuracy of 60% to 65% approx. To achieve accuracy higher than this, we definitely need to research in deep.

Based on all the experiments performed in machine learning algorithms and techniques, input data plays an important role. We are forced to combined the dataset and set of feature list formed accordingly so, where the dataset is divided into testing and training part, the number eventually become very less which is nothing but noise and unwanted information which by using filtering techniques are removed from the dataset to work efficiently and predict the outcome betterwith almost no sign of noise Additionally, SVM has demonstrated that we can generator increasingly

custom list of capabilities and acquire forecasts with incredible proficiency. We have directed tests utilizing non straight RBF portion which is demonstrating extensive precision in result. What's more, the most significant thing, the above investigation helped us in anticipating the future result of costs of organization yet they additionally gave us important and profound bits of knowledge about the idea of information which is positively can be utilized to prepare our SVM classifiers in a superior manner. The venture can be extended further by ad labeling highlight list and with various classifier. Future work should be possible by including the unaided preprocessor use alongside the direct classifier.

Based on the performances of all the three Algorithms, Linear Regression, Support Vector Machine, and Decision Tree, we concluded that Decision Tree is best among the three and second comes the Linear Regression based on the RMSE values,

$$\text{RMSE(DT)} < \text{RMSE(LR)} < \text{RMSE(SVM)}$$

As the RMSE should be least which means the error or difference between the Actual or Y_{test} and Predicted_Y , so it should be less because if they will be less there would be more chances of the two to be close enough and thus a good prediction to be called.

$$\text{RMSE(DT)} - 833.0699$$

$$\text{RMSE(LR)} - 923$$

$$\text{RMSE(SVM)} - 5051.96$$

Also based on the Pie Chart if we calculate the modular difference between Actual and Predicted mean, we would easily conclude that Decision Tree(DT) is the best as it have a modular difference of 35.37, Linear Regression being the second in the race with 76.85 and SVM at the last having 1255.22 mean difference.

5.1 Future scope of improvement

1. Our dataset and analysis method can improve potentially.
2. If more accurate algorithm and refined data with precise research is taken then future scope can be done with possible improvement.
3. Introduction of twitter feeds.
4. Advanced predictions form news feed and different websites can be taken for better results.
5. Refining key phase extraction and doing more work will definitely produce better results.

References

- [1] S. M. Idrees, M. A. Alam and P. Agarwal, "A Prediction Approach for Stock Market Volatility Based on Time Series Data," in *IEEE Access*, vol. 7, pp. 17287-17298, 2019. doi: 10.1109/ACCESS.2019.2895252
- [2] E. W. Saad, D. V. Prokhorov and D. C. Wunsch, "Comparative study of stock trend prediction using time delay, recurrent and probabilistic neural networks," in *IEEE Transactions on Neural Networks*, vol. 9, no. 6, pp. 1456-1470, Nov. 1998. doi: 10.1109/72.728395
- [3] J. Chou and T. Nguyen, "Forward Forecast of Stock Price Using Sliding-Window Metaheuristic-Optimized Machine-Learning Regression," in *IEEE Transactions on Industrial Informatics*, vol. 14, no. 7, pp. 3132-3142, July 2018. doi: 10.1109/TII.2018.2794389
- [4] P. Chang, C. Fan and C. Liu, "Integrating a Piecewise Linear Representation Method and a Neural Network Model for Stock Trading Points Prediction," in *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 39, no. 1, pp. 80-92, Jan. 2009. doi: 10.1109/TSMCC.2008.2007255
- [5] L. Zhang, N. Liu and P. Yu, "A Novel Instantaneous Frequency Algorithm and Its Application in Stock Index Movement Prediction," in *IEEE Journal of Selected Topics in Signal Processing*, vol. 6, no. 4, pp. 311-318, Aug. 2012. doi: 10.1109/JSTSP.2012.2199079
- [6] S. D. Bekiros, "Sign Prediction and Volatility Dynamics With Hybrid Neurofuzzy Approaches," in *IEEE Transactions on Neural Networks*, vol. 22, no. 12, pp. 2353-2362, Dec. 2011. doi: 10.1109/TNN.2011.2169497
- [7] K. Raza, "Prediction of Stock Market performance by using machine learning techniques," 2017 International Conference on Innovations in Electrical Engineering and

Computational Technologies (*ICIEECT*), Karachi, 2017, pp. 1-1.
doi: 10.1109/ICIEECT.2017.7916583

[8] Z. Hu, J. Zhu and K. Tse, "Stocks market prediction using Support Vector Machine," 2013 6th International Conference on Information Management, Innovation Management and Industrial Engineering, Xi'an, 2013, pp. 115-118.
doi: 10.1109/ICIIM.2013.6703096

[9] Z. Wang, S. Ho and Z. Lin, "Stock Market Prediction Analysis by Incorporating Social and News Opinion and Sentiment," 2018 IEEE International Conference on Data Mining Workshops (*ICDMW*), Singapore, Singapore, 2018, pp. 1375-1380.
doi: 10.1109/ICDMW.2018.00195

[10] S. Sarode, H. G. Tolani, P. Kak and C. S. Lifna, "Stock Price Prediction Using Machine Learning Techniques," 2019 International Conference on Intelligent Sustainable Systems (*ICISS*), Palladam, Tamilnadu, India, 2019, pp. 177-181.
doi: 10.1109/ISS1.2019.8907958

[11]<https://scikit-learn.org>, "Support Vector Machine"

[12]a:https://scikit-learn.org/0.18/auto_examples/svm/plot_iris.html

[13]b:<https://theinvestorsbook.com/decision-tree-analysis.html>

