VISVESVARAYA TECHNOLOGICAL UNIVERSITY

JNANA SANGAMA, BELAGAVI – 590 018

A Mini Project Report on

"STOCK PRICE PREDICTION SYSTEM"

Submitted in partial fulfillment of the requirements as a part of the

AI/ML

INTERNSHIP

(NASTECH)

For the award of degree of

Bachelor of Engineering in

Information Science and Engineering

Submitted by

RAASHIL AADHYANTH
1RN18IS081

RAMYA SHETTY
1RN18IS084

Internship Project Coordinators

Dr. R Rajkumar
Associate Professor
Dept. of ISE, RNSIT

Mrs. Sudha V
Assistant Professor
Dept. of ISE, RNSIT

Department of Information Science and Engineering

RNS Institute of Technology

Channasandra, Dr. Vishnuvardhan Road, RR Nagar
Post, Bengaluru – 560 098

2020 -2021

RNS Institute of Technology

Channasandra, Dr. Vishnuvardhan Road,
RR Nagar Post, Bengaluru – 560 098

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that the mini project report entitled ***STOCK PRICE PREDICTION SYSTEM*** has been successfully completed by **RAASHIL AADHYANTH** bearing USN **1RN18IS081** and **RAMYA SHETTY** bearing USN **1RN18IS084** , presently VII semester students of **RNS Institute of Technology** in partial fulfillment of the requirements as a part of the ***AI/ML Internship (NASTECH)*** for the award of the degree of ***Bachelor of Engineering in Information Science and Engineering*** under **Visvesvaraya Technological University, Belagavi** during academic year **2021 – 2022**. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report and deposited in the departmental library. The mini project report has been approved as it satisfies the academic requirements as a part of Internship.

Dr. R Rajkumar	**Mrs. Sudha V**	**Dr. Suresh L**
Coordinator	Guide	Professor and HoD
Associate Professor	Assistant Professor	

External Viva

Name of the Examiners	**Signature with date**
1.	
2.	

ABSTRACT

The real estate sector is an important industry with many stakeholders ranging from regulatory bodies to private companies and investors. Among these stakeholders, there is a high demand for a better understanding of the industry operational mechanism and driving factors. Today there is a large amount of data available on relevant statistics as well as on additional contextual factors, and it is natural to try to make use of these in order to improve our understanding of the industry

House price prediction project focuses on forecasting the coherent house prices for non-house holders based on their financial provisions and their aspirations. By analyzing the foregoing merchandise, fare ranges and also forewarns developments, speculated prices will be estimated. The motive of this paper is to help the seller to estimate the selling cost of a house perfectly and to help people to predict the exact time slap to accumulate a house. Some of the related factors that impact the cost were also taken into considerations such as physical conditions, concept and location etc.

House price prediction on a data set has been done by using linear regression technique. Moreover, this project can be considered as a further step towards more evidence-based decision making for the benefit of these stakeholders. The project focuses on assessment value for residential properties in Calgary between 2017-2020. The aim of our project is to build a predictive model for change in house prices in the year 2021 based on certain time and geography dependent variables.

ACKNOWLEDGMENT

The fulfillment and rapture that go with the fruitful finishing of any assignment would be inadequate without the specifying the people who made it conceivable, whose steady direction and support delegated the endeavors with success.

We would like to profoundly thank **Management** of **RNS Institute of Technology** for providing such a healthy environment to carry out this AI/ML Internship Project.

We would like to express our thanks to our Principal **Dr. M K Venkatesha** for his support and inspired us towards the attainment of knowledge.

We wish to place on record our words of gratitude to **Dr. Suresh L,** Professor and Head of the Department, Information Science and Engineering, for being the enzyme and master mind behind our Internship Project.

We would like to express our profound and cordial gratitude to my Internship Project Coordinators**, Dr. R Rajkumar,** Associate Professor, Department of Information Science and Engineering for their valuable guidance, constructive comments, continuous encouragement throughout the Internship Project and guidance in preparing report.

I would like to express my profound and cordial gratitude to my Faculty Incharge, **Mrs. Sudha V**, Associate Professor, Department of Information Science and Engineering for his/her valuable guidance in preparing Project report.

We would like to thank all other teaching and non-teaching staff of Information Science & Engineering who have directly or indirectly helped us to carry out the Internship Project.

Also, we would like to acknowledge and thank our parents who are source of inspiration and instrumental in carrying out this Mini Project Work.

RAASHIL AADHYANTH
USN: 1RN18IS081

RAMYA SHETTY
USN: 1RN18IS084

TABLE OF CONTENTS

Contents

LIST OF FIGURES

Chapter 1

1. INTRODUCTION

1.1. ORGANIZATION/INDUSTRY

1.1.1. COMPANY PROFILE

NASTECH is formed with the purpose of bridging the gap between Academia and Industry Nastech is one of the leading Global Certification and Training service providers for technical and management programs for educational institutions. We collaborate with educational institutes to understand their requirements and form a strategy in consultation with all stakeholders to fulfill those by skilling, reskilling and upskilling the students and faculties on new age skills and technologies.

1.1.2. DOMAIN/TECHNOLOGY

The domain chosen for our project is AI/ML. Machine learning, the fundamental driver of AI, is possible through algorithms that can learn themselves from data and identify patterns to make predictions and achieve your predefined goals, rather than blindly following detailed programmed instructions, like in traditional computer programming. This technology allows the machine to perceive, learn, reason and communicate through observation of data, like a child that grows up and acquires knowledge from examples. Machines also have the advantage of not being limited by our inherent biological limitations. With machine learning, manufacturing companies have increased production capacity up to 20%, while lowering material consumption rates by 4%.

Nowadays, the revolutionary AI technology evolved from rule-based expert systems to machine learning and more advanced subcomponents such as deep learning (learning representations instead of tasks), artificial neural networks (inspired by animal brains) and reinforcement learning (virtual agents rewarded if they made good decisions).

The AI can master the complexity of the intertwining industrial processes to enhance the whole flow of production instead of isolated processes. This enormous cognitive capacity gives the AI the ability to consider the spatial organization of plants and the timing constraints of live production. Another key advantage is the capability of AI algorithms to think probabilistically, with all the subtlety this allows in edge cases, instead of traditional rule-based methods that require rigid theories and a full comprehension of problems.

1.1.3. Department

R.N.Shetty Institute of Technology (RNSIT) established in the year 2001, is the brain-child of the Group Chairman, Dr. R. N. Shetty. The Murudeshwar Group of Companies headed by Sri. R. N. Shetty is a leading player in many industries viz construction, manufacturing, hotel, automobile, power & IT services and education. The group has contributed significantly to the field of education. A number of educational institutions are run by the R. N. Shetty Trust, RNSIT being one amongst them. With a continuous desire to provide quality education to the society, the group has established RNSIT, an institution to nourish and produce the best of engineering talents in the country. RNSIT is one of the best and top accredited engineering colleges in Bengaluru.

1.2. PROBLEM STATEMENT

1.2.1. Existing System and their Limitations

Prediction of stock trend has long been an intriguing topic and is extensively studied by researchers from different fields. Machine learning, a well-established algorithm in a wide range of applications, has been extensively studied for its potentials in prediction of financial markets. Popular algorithms, including support vector machine (SVM) and reinforcement learning, have been reported to be quite effective in tracing the stock market and help maximizing the profit of stock option purchase while keep the risk low [1-2]. However, in many of these literatures, the features selected for the inputs to the machine learning algorithms are mostly derived from the data within the same market under concern. Such isolation leaves out important information carried by other entities and make the prediction result more vulnerable to local perturbations.

Efforts have been done to break the boundaries by incorporating external information through fresh financial news or personal internet posts such as

Twitter. These approaches, known as sentiment analysis, replies on the attitudes of several key figures or successful analysts in the markets to interpolate the minds of general investors. Despite its success in some occasions, sentiment analysis may fail when some of the people are biased, or positive opinions follow past good performance instead of suggesting promising future markets.

1.2.2. Proposed Solution

In this project, we propose the use of global stock data in associate with data of other financial products as the input features to machine learning algorithms such as SVM. In particular, we are interested in the correlation between the closing prices of the markets that stop trading right before or at the beginning of US markets. As the connections between worldwide economies are tightened by globalization, external perturbations to the financial markets are no longer domestic. It is to our belief that data of oversea stock and other financial markets, especially those having strong temporal correlation with the upcoming US trading day, should be useful to machine learning based predictor, and our speculation is verified by numerical results.

Chapter 2

2. ANALYSIS

FUNCTIONAL REQUIREMENTS

Functional requirements deal with the functionality of the software in the engineering view. The component flow and the structural flow of the same is enhanced and described by it.

The functional statement deals with the raw datasets that are categorized and learning from the same dataset. Later the datasets are categorized into clusters and the impairment of the same is checked for the efficiency purpose. After the dataset cleaning the data are cleansed and the machine learns and finds the pattern set for the same it undergoes various iteration and produce output.

NON-FUNCTIONAL REQUIREMENTS

Non-functional requirement deals with the external factors which are non-functional in nature It is used for analysis purpose. Under the same the judgment, operations are carried out for its performance. Stock is feasible and is ever changing so these extra effects and the requirements helps it to get the latest updates and integrate in a one-go where the technicians can work on and solve a bug or a draft if any.

The non-functional requirements followed are its efficiency and hit gain ratio. The usability of the code for the further effectiveness and to implement and look for the security console. The System is reliable and the performance is maintained with the support of integration and portability of the same.

2.1.HARDWARE REQUIREMENTS

Processor : Intel i5 or above

RAM : Minimum 225MB or
more. Hard Disk: Minimum 2 GB of
space Input Device : Keyboard
Output Device : Screens of Monitor or a Laptop

2.2.SOFTWARE REQUIREMENTS

2.2.1.	Operating system	:	Windows & Linux
2.2.2.	IDE	:	Jupiter Notebook
2.2.3.	Data Set	:	.csv file
2.2.4.	Visualization	:	mat plot lib, pandas.
2.2.5.	Server	:	Web Server with HTTP process.

Chapter 3

3. Design and Implementation

One of the integral parts to maintain the consistency is the literature survey. It’s the crucial steps to be followed in the development process. The Software Development needs authenticity of the resources and the availability of the same. This part helps in discovering the content that been worked on and find the utilization and the implementation of the same in today’s time. The key factor to the development is the economy and the strength of the product. Once the innovation of the same undergoes through the building phase the support and the resource flow is to be monitored and computed. This is also known as the Research phase where all the research is embedded and done to carry the flow.

Machine Learning

One of the finest words heard in today time is Machine Learning. Either it be at work or different places the machine learning has been an integral part of today’s technology. Though its revolutionizing and developing in a rapid rate and development and deployment of the same is still in progress. Machine learning itself has brought a tremendous change in today because of which automation is in frame which was a mere existence in the past.

It's an aspiring term in today’s time. One of the moves that all the firms are interested into. It’s a leading pillar for tomorrow leading the world to a better future of evolution where the customization and labor work can be reduced to half and the safety of the survival can be withheld to stand tall for the better utilization of human mind. Keeping that in picture it’s been a hazard to many more in terms of irrespective field of interest. Since Machine is considered most efficient and the level of mistakes are kept at the minimum the level of work flow can be a work of hazard and further improvement on the same may create thousands sitting idle in home creating a larger impact on unemployment and livelihood. Which in other way is a threat to the society too.

Though its being evolutional in nature but it has integrated itself well with the terms of computational and digitalization. Various computational fields like Data Mining, Statistical Analysis, Optimization of resources, Automation are a major part of it. Here the machine has the capacity to process the result on its own as same as the human bring. This process can be initiator as well as the derivable. The statistical flow is mainly reasonable with data driven pattern even the unstructured or the semi-structured data can be processed and approximate answer to the same can be derived.

closest value to its aligned field is found and the proximity is determined. The classification of the same can be listed as follows:

A. Supervised Learning

Supervised Learning deals with the supervision of the machine to derive the necessity input required. It’s a mathematical model where the inputs and output of the same is already known and its passed to the machine to get the expected output so that the efficiency is determined and this is the learning phase for the machine. Here the feeding and derivation of the same is measured.

Here the machine filters the inputs learns from the functional unit. Compute it and stores it into its memory for further process and if found a matching pattern it uses the same and learns from it and plot a result out of the same.

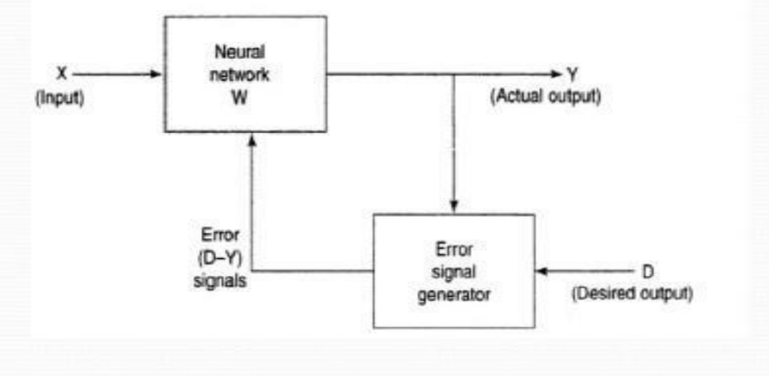


Figure 1: Supervised learning

This is a dependent process. The machine totally depends on the user who has to feed the inputs and has to check the efficiency of the same and correct it with the flow of iteration. It’s an ANN network. During the training phase vectors are taken into consideration.

Up in the above figure There’s an input vector and the output vector. The input vector derives and gives an output flow of the output vector. If the error signal is generated then the iteration is undergone where as lacking of the same means the

B. Unsupervised Learning

Unsupervised learning deals with learning by itself. It is also known as self-learning algorithm. Here only the input vector is known and passed. Thus, the variance of the result deals with the input factors. HTest Data are passed and with the iteration of the same it learns from it derives itself much closer to the conclusion part. Labelled is missed in the data set and classification and categorization of the same had to be done the machine itself. Cluster and Communalization is the main essence of it.

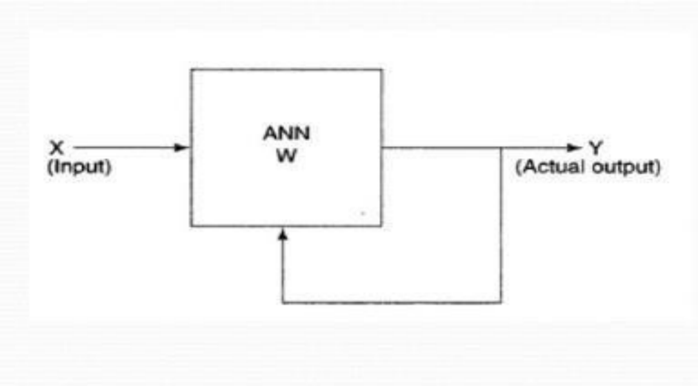


Figure 2:Unsupervised Learning

As described in the above figure, in this ANN network when the input is processed by the function the output had to be derived and to be matched with the cluster set to provide the result. If the result lacks interpretation, then it undergoes the iteration. All the data sets are formed and combined in a cluster set for the effective uses of the same in further cases.

C. Reinforcement Learning

In this type of learning a reinforced strategy is used. Its deals with blooming of the knowledge. It's neither Supervised nor Unsupervised form of learning. They use dynamic techniques for letting the user know the output and the derivation of the same.

In these sorts of algorithms, they don't assume the environmental set. These are even used in higher and complex mechanism finding likes genetic algorithm. They are widely in

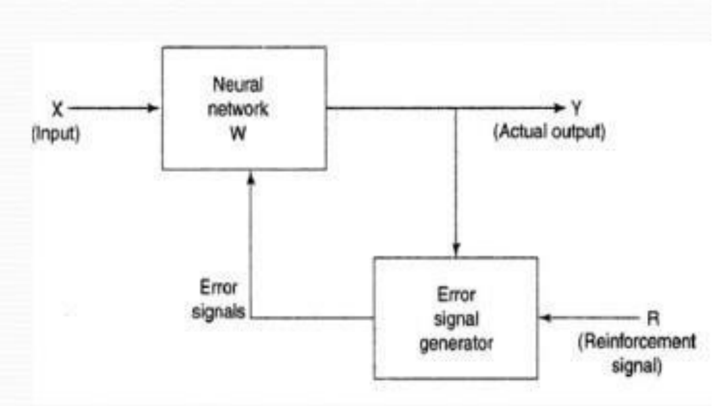


Figure 3:Reinforcement Learning

progress and implemented most in automation for the better efficiency of the establishment. These algorithms are used in Games and Automation of the vehicle resources.

As described in the figure the input vector is passed to a ANN model where the functionalities of the same are stored. If the accurate output is derived then a reward is given to the user making it go to the next level for further task of completion. If not, then the Error signal is generated for the same. The accuracy level is calculated and passed down to the user stating the same.

DESIGN

3.1. Structure Chart

A structure chart (SC) in software engineering and organizational theory is a chart which shows the breakdown of a system to its lowest manageable levels. They are used in structured programming to arrange program modules into a tree. Each module is represented by a box, which contains the module's name.

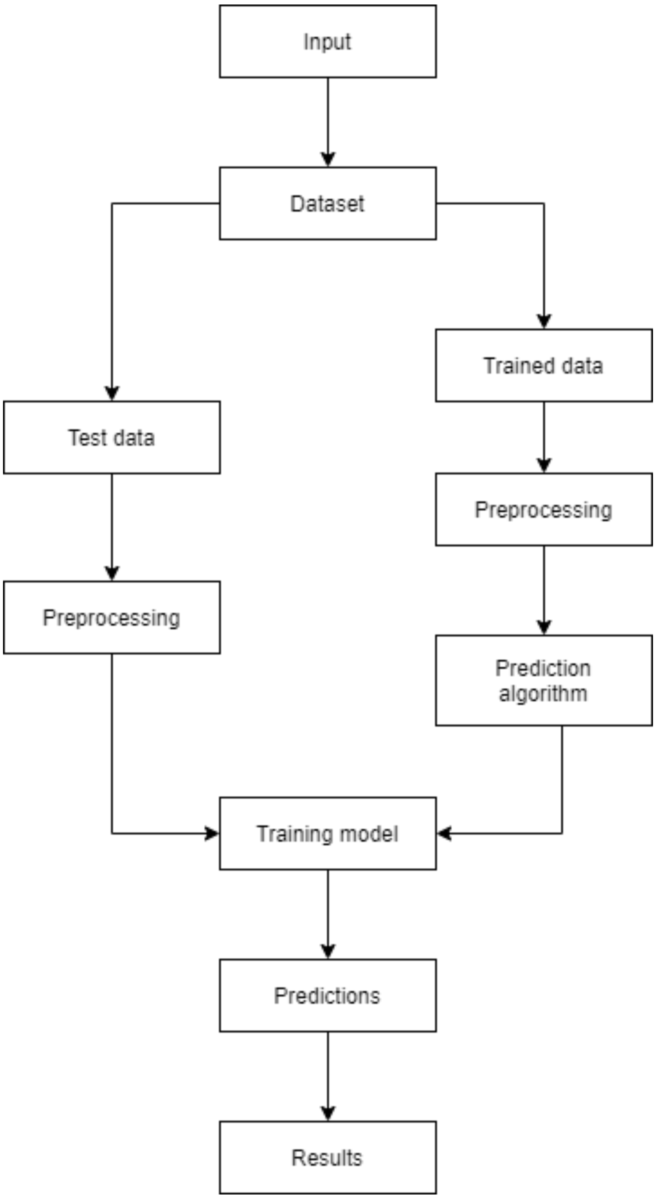


Figure 4:Training and prediction

3.1.1. UML Diagrams

A UML diagram is a partial graphical representation (view) of a model of a system under design, implementation, or already in existence. UML diagram contains graphical elements (symbols) - UML nodes connected with edges (also known as paths or flows) - that represent elements in the UML model of the designed system. The UML model of the system might also contain other documentation such as use cases written as templated texts.

The kind of the diagram is defined by the primary graphical symbols shown on the diagram. For example, a diagram where the primary symbols in the contents area are classes is class diagram. A diagram which shows use cases and actors is use case diagram. A sequence diagram shows sequence of message exchanges between lifelines.

UML specification does not preclude mixing of different kinds of diagrams, e.g., to combine structural and behavioral elements to show a state machine nested inside a use case. Consequently, the boundaries between the various kinds of diagrams are not strictly enforced. At the same time, some UML Tools do restrict set of available graphical elements which could be used when working on specific type of diagram.

UML specification defines two major kinds of UML diagram: structure diagrams and behavior diagrams.

Structure diagrams show the static structure of the system and its parts on different abstraction and implementation levels and how they are related to each other. The elements in a structure diagram represent the meaningful concepts of a system, and may include abstract, real world and implementation concepts.

Behavior diagrams show the dynamic behavior of the objects in a system, which can be described as a series of changes to the system over time.

3.1.2. Use Case Diagram

In the Unified Modelling Language (UML), a use case diagram can summarize the details of your system's users (also known as actors) and their interactions with the system. To build one, you'll use a set of specialized symbols and connectors. An effective use case diagram can help your team discuss and represent:

- Scenarios in which your system or application interacts with people, organizations, or external systems.
- Goals that your system or application helps those entities (known as actors) achieve.
- The scope of your system.

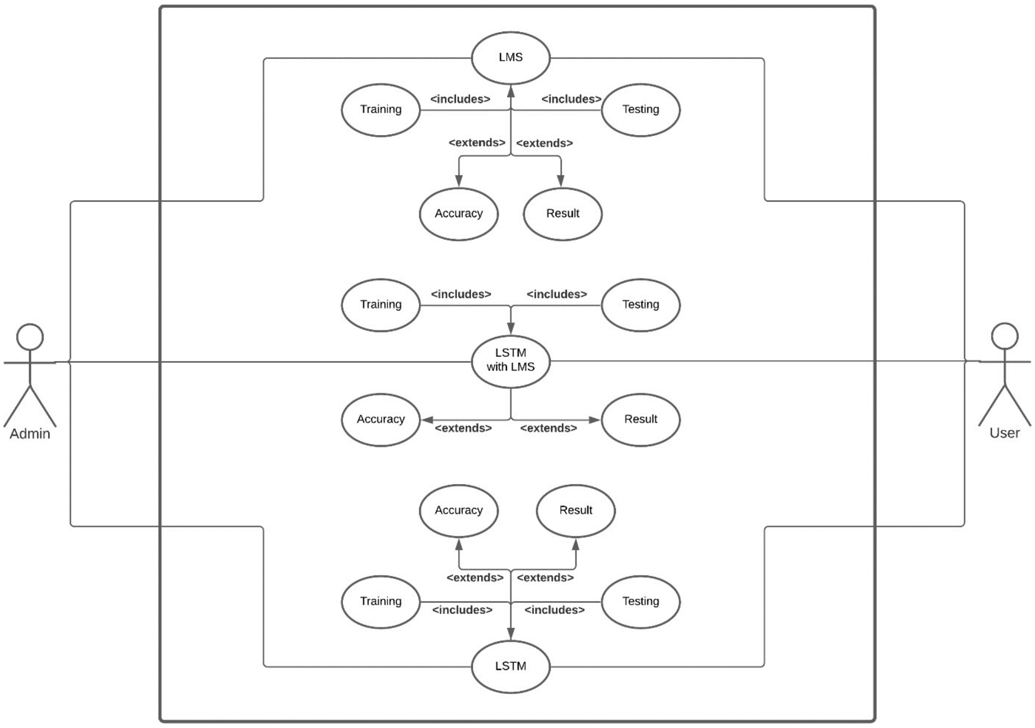


Figure 5:Using LMS, LSTM and LSTM with LMS in the system

3.1.3. Sequence Diagram

A sequence diagram is a type of interaction diagram because it describes how and in what order a group of objects works together. These diagrams are used by software developers and business professionals to understand requirements for a new system or to document an existing process. Sequence diagrams are sometimes known as event diagrams or event scenarios.

Sequence diagrams can be useful references for businesses and other organizations. Try drawing a sequence diagram to:

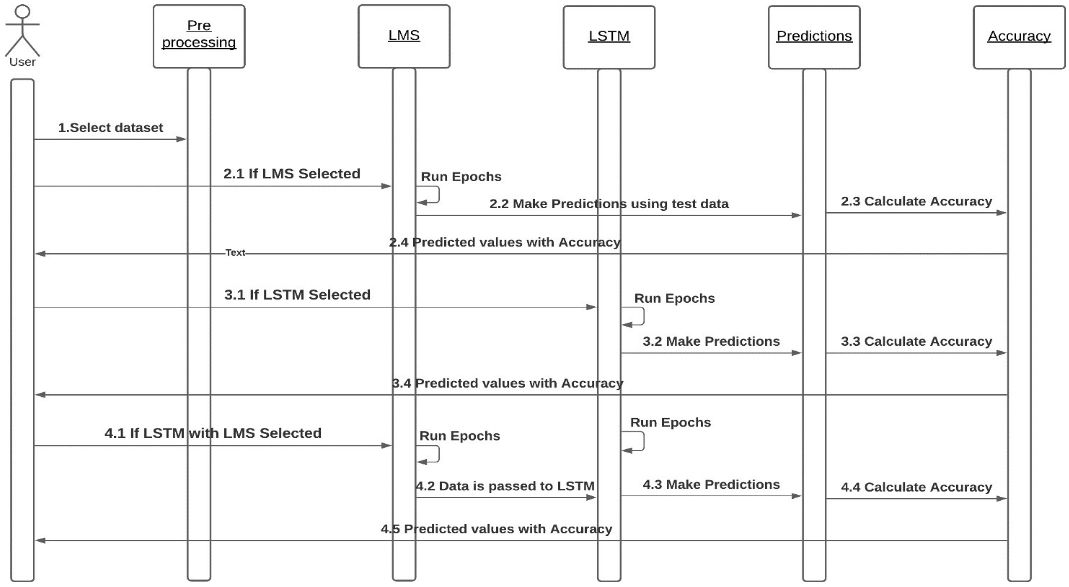


Figure 6:Execution based on model selection

3.1.4. Flow Chart

A flowchart is a type of diagram that represents a workflow or process. A flowchart can also be defined as a diagrammatic representation of an algorithm, a step-by-step approach to solving a task. The flowchart shows the steps as boxes of various kinds, and their order by connecting the boxes with arrows.

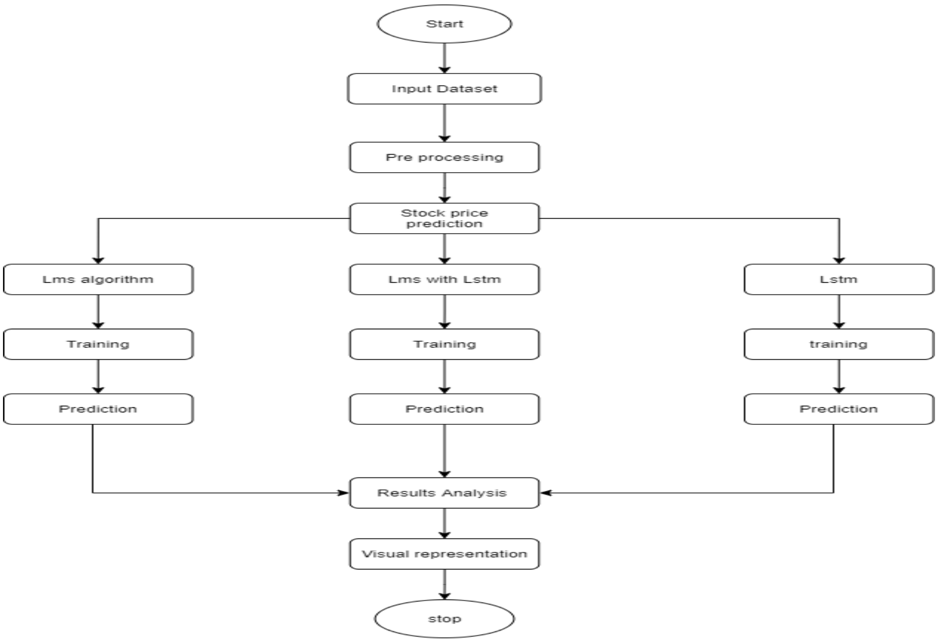


Figure 7:Flow of execution

3.1.5. Component Diagram

Component diagram is a special kind of diagram in UML. The purpose is also different from all other diagrams discussed so far. It does not describe the functionality of the system but it describes the components used to make those functionalities.

Component diagrams are used in modeling the physical aspects of object-oriented systems that are used for visualizing, specifying, and documenting component-based systems and also for constructing executable systems through forward and reverse engineering. Component diagrams are essentially class diagrams that focus on a system's components that often used to model the static implementation view of a system.

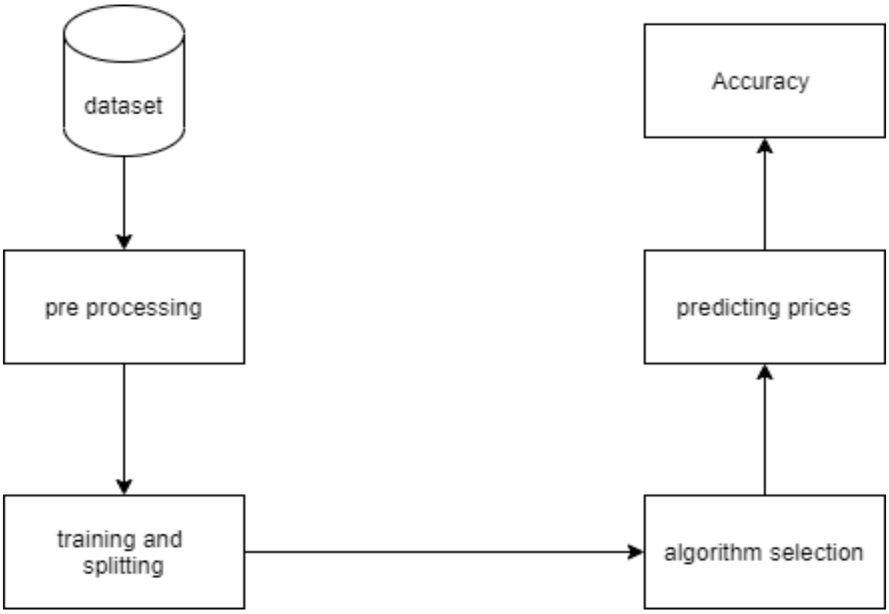


Figure 8:Components present in the system

IMPLEMENTATION

3.2. Data Collection and algorithm

The training data used in our project were collected from Bloomberg Database. In this project, we picked 3M Stock to apply our method. The data contains daily stock information ranging from 1/9/2008 to 11/8/2013 (1471 data points). There are 16 features that we can use to apply our learning theory. In addition, we used the daily labeling as follows: label "1" if the closing price is higher than that of the previous day. Otherwise, label "-1". For example, if the closing price of stock A on 11/11/2013 is higher than that on 11/10/2013, and on 11/10/2013, the PE ratio, PX volume, PX ebitda,S&P 500 index are X1, X2,...,X15, so the training data of A on 11/10/2013 is (X, Y), where X = (X1, X2,...,X15)T, Y = (+1)

Stock	3M Co (NYSE: MMM)
Features	PE ratio, PX volume, PX ebitda, current enterprise value, 2-day
	net price change, 10-day volatility, 50-day moving average, 10-day moving average, quick ratio, alpha overridable, alpha for beta pm, beta raw overridable, risk premium, IS EPS, and corresponding S&P 500 index
Data Source	Bloomberg Data Terminal

3.2.1. Model selection

3.2.1.1.1. Next-Day Model

In our project, we mainly applied supervised learning theories, i.e. Logistic Regression, Gaussian Discriminant Analysis, Quadratic Discriminant Analysis, and SVM. The most important result that we should watch closely is the accuracy of prediction, which we define as follows:

$$Accuracy\ = \frac{number\ of\ days\ that\ the\ model\ classified\ testing\ data}{total\ number\ of\ testing\ days}$$

We used 70% of the data sets as training data and tested our fitted models with the remaining 30% data sets.

Model	Logistic Regression	GDA	QDA	SVM
Accuracy	44.5%	46.4%	58.2%	55.2%

It turned out that the next-day prediction has a very low accuracy with the highest accuracy (QDA) being only 58.2%. We know that by flipping a coin we can probably get an accuracy of roughly 50% since the investing decision is binomial. Such result can be explained by the semi-strong efficient market theory, which states that all public information is calculated into a stock's current share price, meaning that neither fundamental nor technical analysis can be used to achieve superior gains.

3.2.1.1.2. Long-Term Model

Although our next-day prediction isn't very positive, we believe the financial data of a particular stock can still provide some insights for the stock's future movement. After all, that' why so many financial institutions/individual investors believe their work is meaningful.

Especially, we think sometimes because of the existence of market sentiment, some information will not be reflected in the stock price immediately. Besides, in the eyes of investors, we also care about the predictions results of longer term to design our long-term investment strategy.

Here, we define our problem as predicting the sign of difference between tomorrow's stock price and that of certain days ago. Again, we used 70% of the data set as training data and tested our fitted models with the remaining 30% data sets.

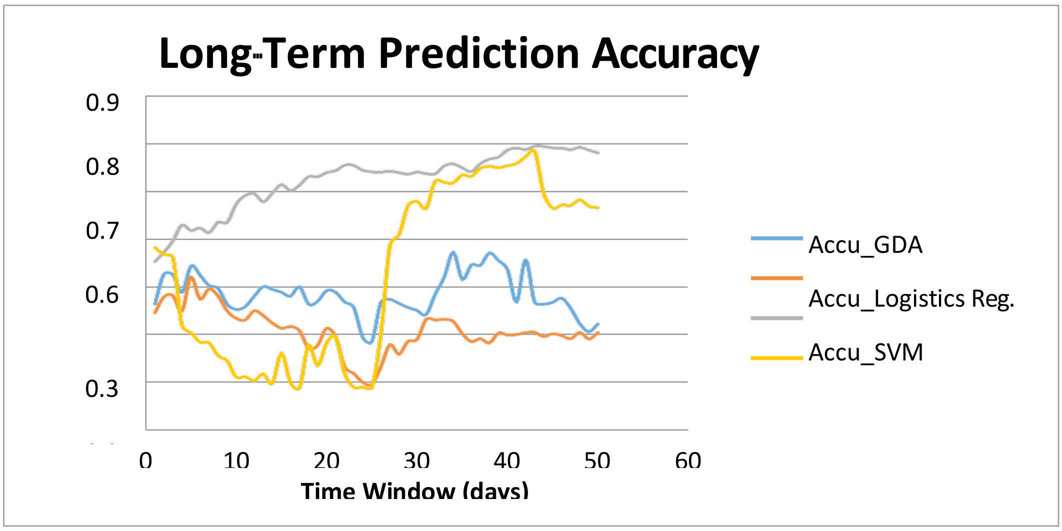


Figure 9:Long term analysis

From the chart, we can see that for SVM and QDA model, the accuracy increases when the time window increases. Furthermore, SVM gives the highest accuracy when the time window is 44 days (79.3%). It's also the most stable model.

3.2.1.1.3. Feature Selection

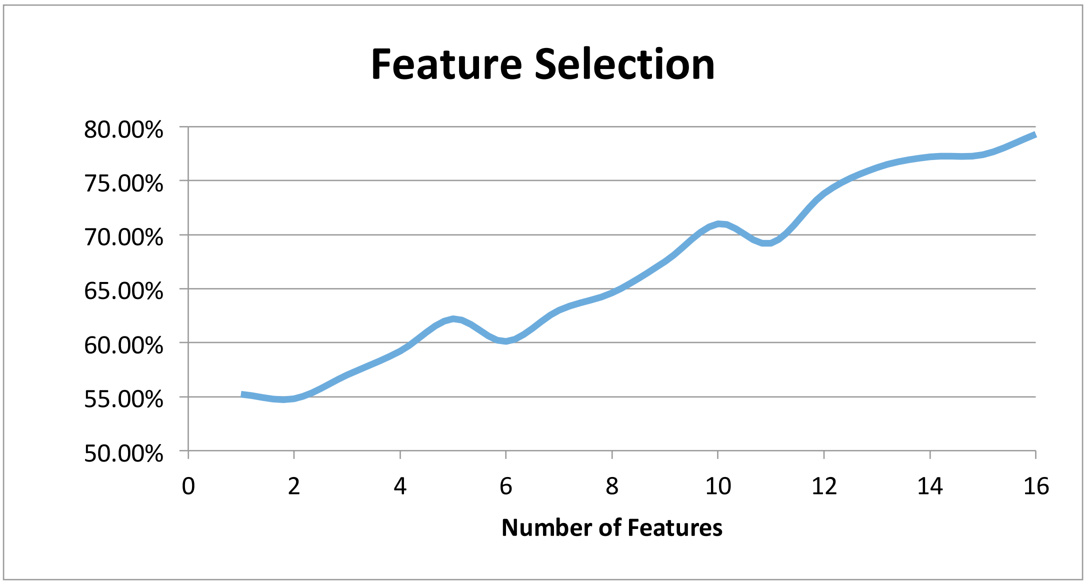


Figure 10:Feature Selection

From the chart established by backward stepwise feature selection, we can see that when we used all of the 16 features, we get our highest prediction accuracy. It makes sense because with over 1400 data points but only 16 features, there's no need to reduce the number of features.

3.2.2. Trading Strategy

3.2.2.1.1. Predictor Characteristics

From the previous analysis, we have already determined that the best predicting model for 3M stock is SVM model. Here we use SVM as our predictor in order to develop our trading strategy.

Predictor	SVM
Kernel	Polynomial
Number of Features	16
Time Window	44 ays

3.2.2.1.2. Strategy Implementation

Initially, we used 990 of our 1470 data points to fit our model. Then we used our model to predict the stock price and made according investment decision on an on-time basis, meaning we will take in new information and update our predictor every trading date. Our back-testing of the strategy is over the course of December 2011 to October 2013.

On each day of the beginning 44 days, we will make a decision whether to buy the stock or not based on our prediction of whether the stock price would go up after 44 days. After the first 44 days, on each day we will make an investment decision again. It's better illustrated in the following decision tree:

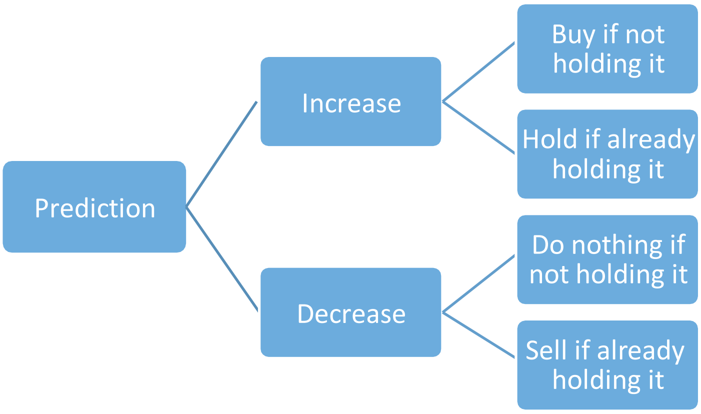

Prediction
Increase
Decrease
Buy if not holding it
Hold if already holding it
Do nothing if not holding it
Sell if already holding it

Figure 11:decision tree

Equivalently, we can interpret the strategy as if there are 44 traders. Trader i is responsible for trading his portfolio on i, 44+i, 44n + i, day. Traders are independent to each other.

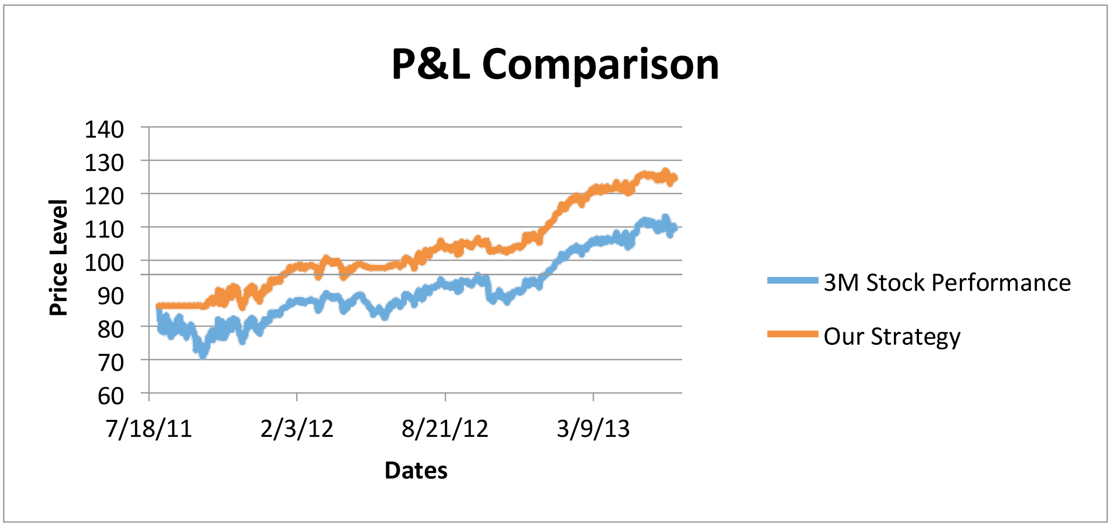


Figure 12:P&L comparison

From the plot above, it's obvious that our strategy has outrun the performance of the stock, with an annualized return 19.3% vs. 12.5%.

3.3. Pseudocode

A. Plot Graph

```
import numpy as np
import matplotlib.pyplot as plt

def dataset_plot(df, symbol):
    x = df.index
    y = df.loc[:, 'Adj. Close']
    plt.plot(x, y)
    plt.xlabel('Dates')
    plt.ylabel('Adj. Close')
    plt.title(f'Adj. Close Price trend for {symbol}')
    plt.show()
    plt.savefig('dataset_plot.png')

def feature_plot(df):
    x = df.index
    y_mdav5 = df.loc[:, 'MDAV5']
    y_macd = df.loc[:, 'MACD']
    y_macd_sline = df.loc[:, 'MACD_SignalLine']

    plt.subplot(3,1,1)
    plt.plot(x, y_mdav5)
    plt.title('MDAV5')

    plt.subplot(3,1,2)
    plt.plot(x, y_macd)
    plt.title('MACD')

    plt.subplot(3,1,3)
    plt.plot(x, y_macd_sline)

    plt.title('MACD_SignalLine')
    plt.show()
    plt.savefig('feature_plot.png')
```

B. Plot Graph

```
import quandl
import datetime
import pandas as pd
import sys
quandl.ApiConfig.api_key = 'SLH4M7i2DCfVc7Npr_zV'

def get_data_from_quandl(symbol, start_date, end_date):
    file_name = symbol + '.csv'
    data = quandl.get("WIKI/"+symbol,
    returns='pandas',
    start_date=start_date,
    end_date=end_date,
    collapse='daily',
    order='asc',
    )
    data.to_csv('datasets/' + file_name)

start_date = '2002-01-01'
end_date = datetime.datetime.today().strftime('%Y-%m-%d')

symbols = sys.argv[1:]

for symbol in symbols:
    print('Getting Data from quandl between ' + start_date + ' and ' +
end_date + ' for the stock ' + symbol)
    get_data_from_quandl(symbol, start_date, end_date)
```

C. Plot Graph

```
from sklearn import svm
import preprocess_data
from sklearn.ensemble import RandomForestClassifier as rfc
from sklearn.ensemble import AdaBoostClassifier as abc
from sklearn.ensemble import VotingClassifier


def train(df):
    '''This function trains the data on 4 different SVC model kernels:
    1. Linear Kernel
    2. Polynomial Kernel
    3. Radial Basis Function Kernel
    4. Sigmoid Kernel
    The RFC model is also implemented.

    The hyperparameters are set default in each case.
    The score of the model on the Dev/Test set is returned to the main
script.
    '''
    X, y = preprocess_data.addFeatures(df)
    X_train, X_test, y_train, y_test = preprocess_data.splitDataset(X, y)
    X_train, X_test = preprocess_data.featureScaling(X_train, X_test)

    model_slinear = svm.SVC(kernel='linear')
    model_slinear.fit(X_train, y_train)
    score_slinear = model_slinear.score(X_test, y_test)

    model_spoly = svm.SVC(kernel='poly')
    model_spoly.fit(X_train, y_train)
    score_spoly = model_spoly.score(X_test, y_test)

    model_srbf = svm.SVC(kernel='rbf')
    model_srbf.fit(X_train, y_train)
    score_srbf = model_srbf.score(X_test, y_test)

    model_ssig = svm.SVC(kernel='sigmoid')
    model_ssig.fit(X_train, y_train)
    score_ssig = model_ssig.score(X_test, y_test)

    model_rfc = rfc(max_depth=4, random_state=0)
    model_rfc.fit(X_train, y_train)
    score_rfc = model_rfc.score(X_test, y_test)

    model_abc = abc(n_estimators=500)
    model_abc.fit(X_train, y_train)
    score_abc = model_abc.score(X_test, y_test)

    model_vc = VotingClassifier(estimators=[('svc', model_srbf,), ('rf', model_rfc)], voting='hard')
    model_vc.fit(X_train, y_train)
    score_vc = model_vc.score(X_test, y_test)

    return score_slinear, score_spoly, score_srbf, score_ssig, score_rfc, score_abc,  score_vc
```

3.4. Technical Survey

3.4.1. SURVEY – I

Historic data are of great values and that been proved by Sathik and Sekhar[1]. They derived a hidden patterns from the dataset and have out generated a investment decision plan using different data mining technologies. They used the same output to invest on the stocks. The efficiency of the same was found to be 84.26% which was consider to be a higher hit rate.

3.4.2. SURVEY – II

ANN or Artificial neural networks was discovered later Liam and Jing[2]. They used the ANN techniques to classify, predict and recognize the data sets. In neural network the brain phenomenon is studied and the implementation of brain neurons are tried to be practiced. Output generated from the same were used in trading prediction and stability. In the research pages they have mentioned a seven prediction models in neural network for the higher efficiency yield. Sampling. Training and recommending are one of it's features mentioned

3.4.3. SURVEY – III

Neural Network was found well integrating with Linear Equation and it's relation. Kun Huang and Tiffany [3] used the same to implement a time series fuzzy network model to improvise and predict the forecasting. The efficiency of the model was found to be deliberate but the computational time was higher than the expected causing it slow for prediction.

3.4.4. SURVEY IV

Bajkunthu and Md. Rafiul [4] approached interrelated market forecasting. The approach for the same was initiated with the help of HMM (Hidden Markov Models). HMM is used for classification of the item set in bulk and can be even help in pattern matching. A hybrid model implemented for efficiency in forecasting of stock market.

Chapter 4

4. Observation and Results

4.1.DESIGN GOALS

To make the project runs smoothly it's required that we make plan and design some accepts like flowcharts and system architecture which are defined below.

4.1.1. Data Collection

Data collection is one of the important and basic things in our project. The right dataset must be provided to get robust results. Our data mainly consists of previous year or weeks stock prices. We will be taking and analyzing data from Kaggle. After that seeing the accuracy, we will use the data in our model.

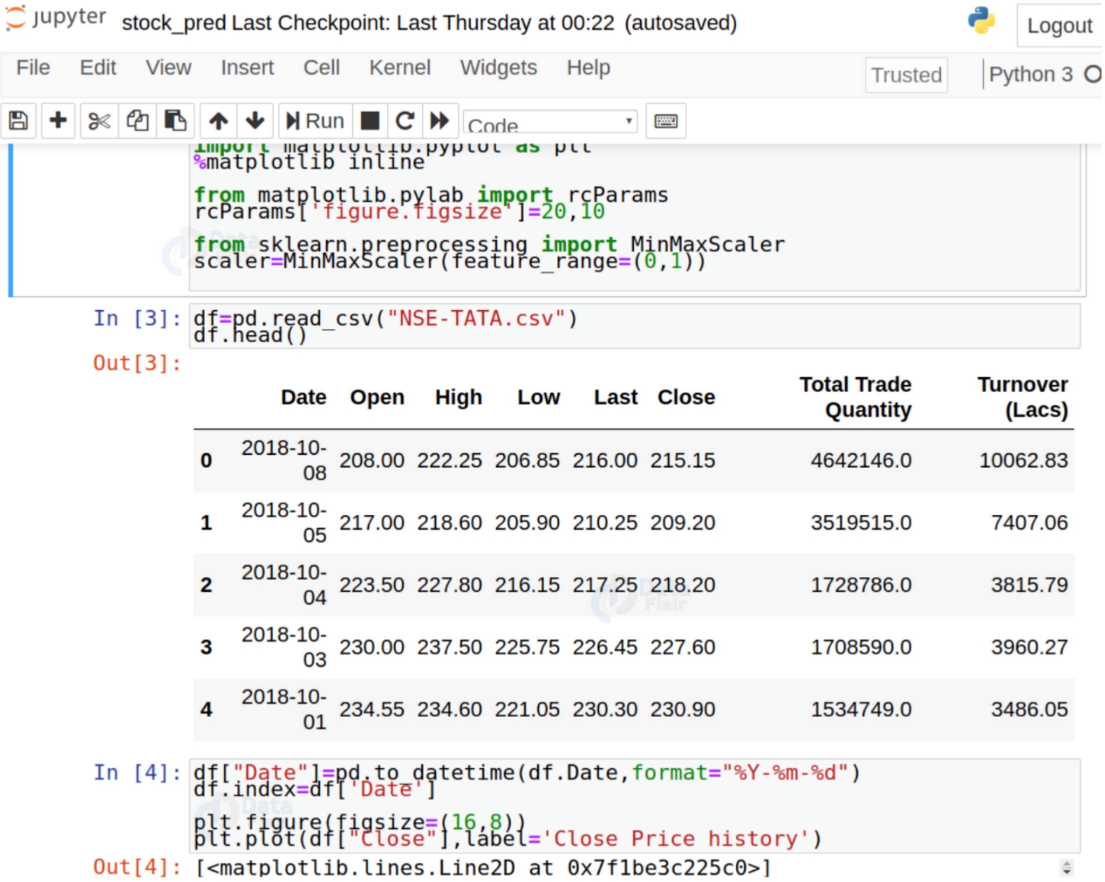
4.1.2. Data Preprocessing

Human can understand any type of data but machine can't our model will also learn from scratch so it's better to make the data more machine readable. Raw data is usually inconsistent or incomplete. Data preprocessing involves checking missing values, splitting the dataset and training the machine etc.

4.1.3. Training Model

Similar to feeding somethings, machine/model should also learn by feeding and learning on data. The data set extracted from Kaggle will be used to train the model. The training model uses a raw set of data as the undefined dataset which is collected from the previous fiscal year and from the same dataset a refine view is presented which is seen as the desired output. For the refining of the dataset various algorithms are implemented to show the desired output.

4.2.Results and Snapshots



```
import matplotlib.pyplot as plt
%matplotlib inline

from matplotlib.pylab import rcParams
rcParams['figure.figsize']=20,10

from sklearn.preprocessing import MinMaxScaler
scaler=MinMaxScaler(feature_range=(0,1))
```

```
In [3]: df=pd.read_csv("NSE-TATA.csv")
        df.head()
```

Out[3]:

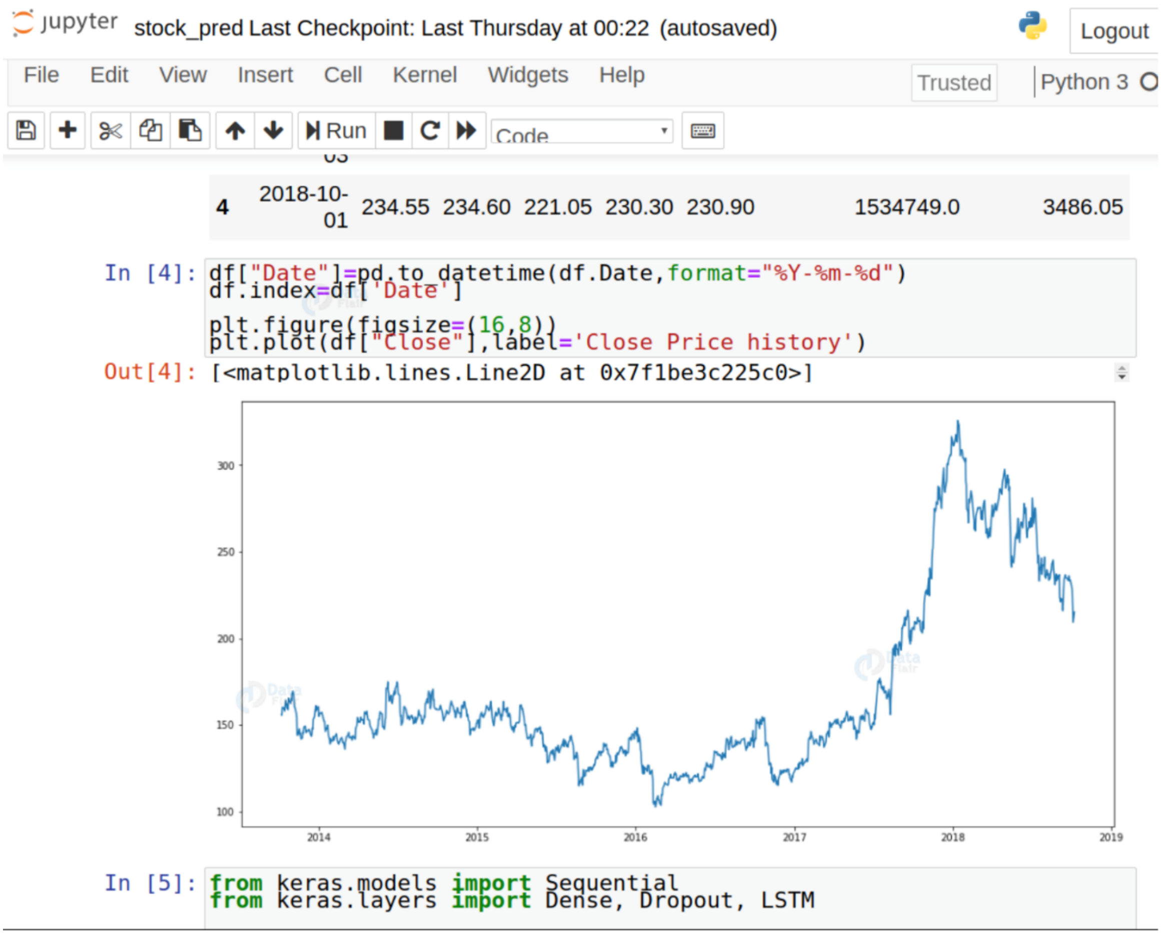
	Date	Open	High	Low	Last	Close	Total Trade Quantity	Turnover (Lacs)
0	2018-10-08	208.00	222.25	206.85	216.00	215.15	4642146.0	10062.83
1	2018-10-05	217.00	218.60	205.90	210.25	209.20	3519515.0	7407.06
2	2018-10-04	223.50	227.80	216.15	217.25	218.20	1728786.0	3815.79
3	2018-10-03	230.00	237.50	225.75	226.45	227.60	1708590.0	3960.27
4	2018-10-01	234.55	234.60	221.05	230.30	230.90	1534749.0	3486.05

```
In [4]: df["Date"]=pd.to_datetime(df.Date,format="%Y-%m-%d")
        df.index=df['Date']

        plt.figure(figsize=(16,8))
        plt.plot(df["Close"],label='Close Price history')

Out[4]: [<matplotlib.lines.Line2D at 0x7f1be3c225c0>]
```

Figure 13:This is to read the dataset



4	2018-10-01	234.55	234.60	221.05	230.30	230.90	1534749.0	3486.05

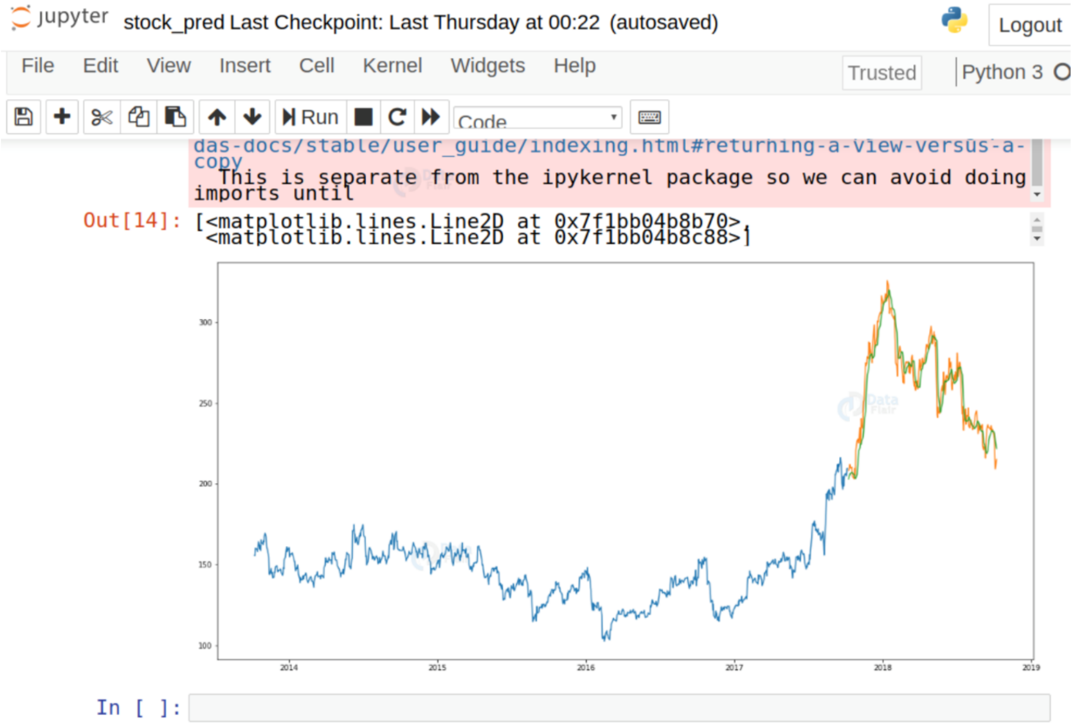
```
In [4]: df["Date"]=pd.to_datetime(df.Date,format="%Y-%m-%d")
        df.index=df['Date']

        plt.figure(figsize=(16,8))
        plt.plot(df["Close"],label='Close Price history')
```

```
Out[4]: [<matplotlib.lines.Line2D at 0x7f1be3c225c0>]
```

```
In [5]: from keras.models import Sequential
        from keras.layers import Dense, Dropout, LSTM
```

Figure 14: Analyzing the closing prices from data frame



```
das-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-
copy
  This is separate from the ipykernel package so we can avoid doing
imports until
```

```
Out[14]: [<matplotlib.lines.Line2D at 0x7f1bb04b8b70>,
          <matplotlib.lines.Line2D at 0x7f1bb04b8c88>]
```

Figure 15:Visualizing the predicted stock cost with actual cost

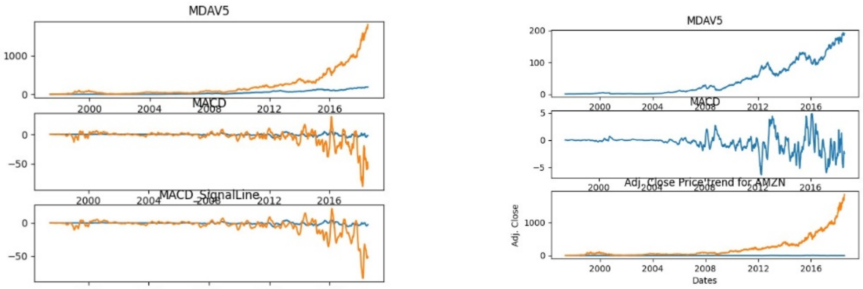


Figure 16: Results of the stock

Chapter 5

5. CONCLUSION AND FUTURE ENHANCEMENT

5.1. Conclusion

In this project, we applied supervised learning techniques in predicting the stock price trend of a single stock. Our finds can be summarized into three aspects:

1. Various supervised learning models have been used for the prediction and we found that SVM model can provide the highest predicting accuracy (79%), as we predict the stock price trend in a long-term basis (44 days).
2. Our feature selection analysis indicates that when use all of the 16 features, we will get the highest accuracy. That's because the number of data points is much bigger than that of the features.
3. The trading strategy based on our prediction achieves very positive results by significantly outrunning the stock performance.

5.2. Future Enhancement

As for our future work, we believe we can make the following improvements:

1. Test our predictor on different stocks to see its robustness. Try to develop a "more general" predictor for the stock market.
2. Construct a portfolio of multiple stocks in order to diversify the risk. Take transaction cost into account when evaluating strategy's effectiveness

Chapter 6

1. REFERENCE

[1] K. Senthamarai Kannan, P. Sailapathi Sekar, M.Mohamed Sathik and P. Arumugam, “Financial stock market forecastusing data mining Techniques", 2010, Proceedings of the international multiconference of engineers and computer scientists.

[2] Tiffany Hui-Kuang yu and Kun-Huang Huarng, “A Neural network-based fuzzy time series model to improve forecasting”, Elsevier, 2010, pp: 3366-3372.

[3] Md. Rafiul Hassan and Baikunth Nath, “Stock Market forecasting using Hidden Markov Model: A New Approach”, Proceeding of the 2005 5th International conference on intelligent Systems Design and Application 0-7695-2286-06/05, IEEE 2005.

[4] Bonde, Ganesh, and Rasheed Khaled. “Extracting the best features for predicting stock prices using machine learning.” Proceedings on the International Conference on Artificial Intelligence (ICAI). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), 2012.

[5] P. Hajek, Forecasting Stock Market Trend using Prototype Generation Classifiers, WSEAS Transactions on Systems, Vol.11, No. 12, pp. 671-80, 2012.

[6] Hagenau, Michael, Michael Liebmann, Markus Hedwig, and Dirk Neumann. "Automated news reading: Stock price prediction based on financial news using context- specific features." InSystem Science (HICSS), 2012 45th Hawaii International Conference on, pp. 1040-1049. IEEE, 2012.

[7] Kyoung-jae Kim, Ingoo Han. “Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index”. Expert Systems with Applications, Volume 19, Issue 2, 2000, Pages 125-132, ISSN 0957-4174.

[8] Leung, Carson Kai-Sang, Richard Kyle MacKinnon, and Yang Wang. “A machine learning approach for stock price prediction.” Proceedings of the 18th International Database Engineering & Applications Symposium. ACM, 2014.

[9] Bonde, Ganesh, and Rasheed Khaled. “Extracting the best features for predicting stock prices using machine learn- ing.” Proceedings on the International Conference on Artificial Intelligence (ICAI). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), 2012.