**19CSE305 – Machine Learning**

**Phase 2**

**P Teja Venkata Subbareddy – AM.EN.U4CSE19341, S5, CSE D**

1. **Problem Definition**

The stock market is known for its extreme complexity and volatility, and people are always looking for an accurate and effective way to predict future stock prices. Different efforts have been done to use different types of Machine learning algorithms for prediction of stock prices. Predicting how the stock market will perform is one of the most difficult things to do. There are so many factors involved in the prediction – physical factors vs. psychological, rational and irrational behavior, etc. All these aspects combine to make share prices volatile and very difficult to predict with a high degree of accuracy.

1. **Datasets.**

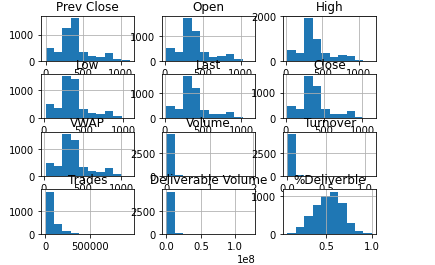
The datasets I chose of companies Cipla, Dreddy’ s, Bharti Airtel daily stock values from the last two decades.

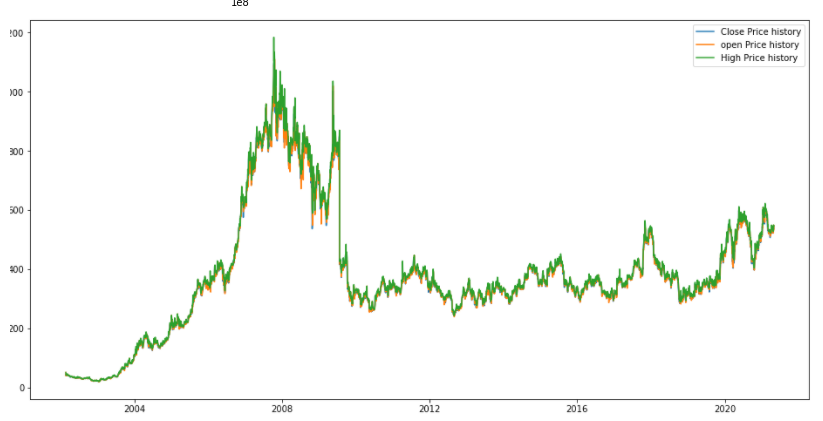
These datasets contain information about the previous close, open, high, last, Close, VWAP values of day and Volume of stocks, Turn over.

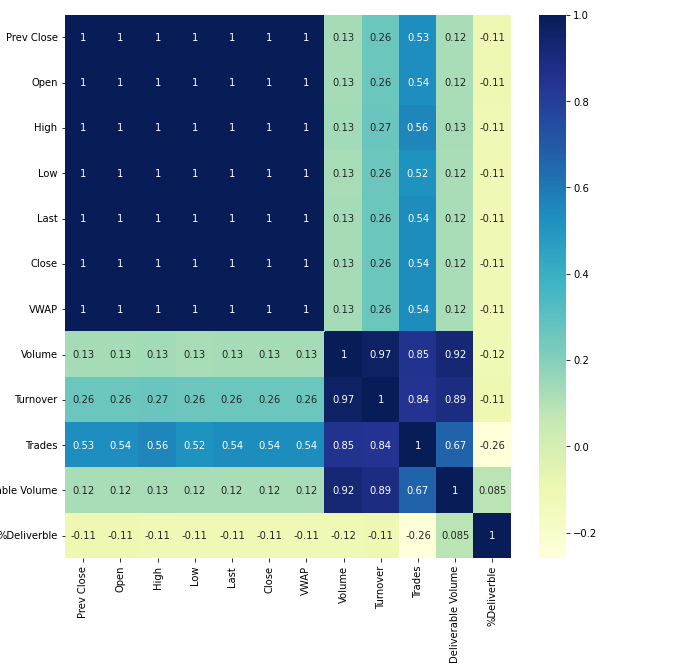
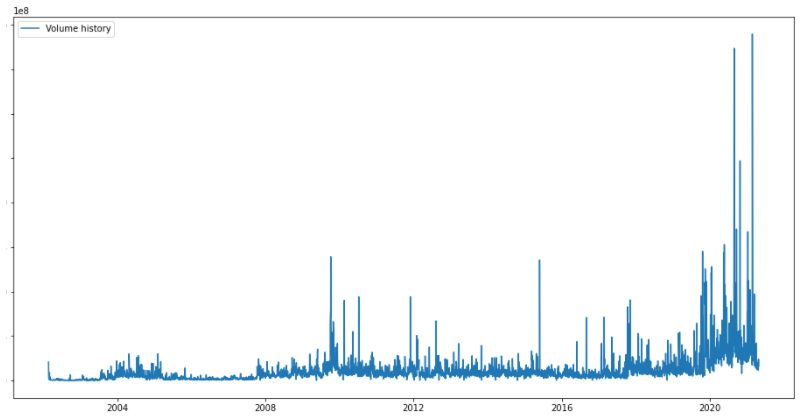
1. **Data Preparation.**

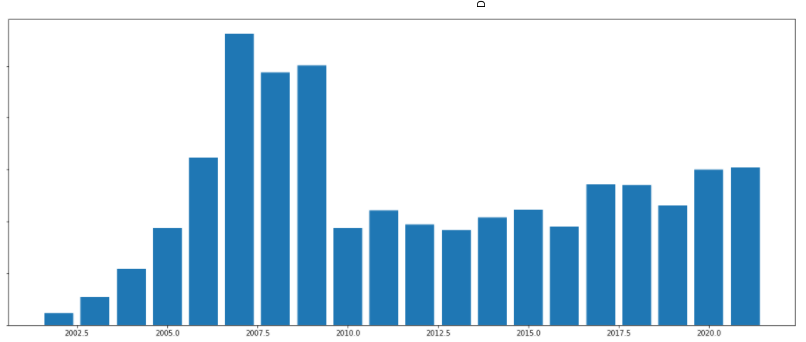
* **Pre processing**

We prepare the dataset clearly set for the model, firstly checking the columns and understanding the type and deciding the situation of the column, then check incomplete data, if any incomplete we are filling the position with mean value of the column. then making the new data with required features. In the dataset the type of date data is in string so the data is remormating to data and breaking the date acc to ‘Year’, ‘Month’, ‘Week’, ‘Day’, ‘Dayofweek’, ‘Dayofyear’, ‘Is\_month\_end’, ‘Is\_month\_start’, ‘Is\_quarter\_end’, ‘Is\_quarter\_start’, ‘Is\_year\_end’, and ‘Is\_year\_start’. so easy to understand the date. and then we visualizing the data so we can get clear understanding about the data columns and relation, using the scatterplot, histograms, box plot we find the relations so we are able to get features about the data.









1. **Packages used**

import csv

Importing CSV library for analyzing and get data from csv files

import fastai

from fastai.tabular import add\_datepart

using to break down the date for simplifying the period and easy to pre - process

import numpy as np

import pandas as pd

**using numpy** we can easily facilitates efficient numerical operations on large quantities of data. The main data structure in this library is the powerful NumPy array, ndarray, which can have any number of dimensions. The NumPy library contains many useful features for performing mathematical and logical operations on these special arrays. NumPy is a part of a set of Python libraries that are used for scientific computing due to its efficient data analysis capabilities.

**using pandas** data manipulation tools that are built on top of and add to those of the established NumPy library. It relies on the NumPy array structure for implementation of its objects and therefore shares many features with NumPy and is frequently used alongside it. Pandas is also a part of the set of libraries used for scientific computation.

import matplotlib.pyplot as plt -> using for plotting graph

import seaborn as sns -> scatter plot

from pandas.plotting import scatter\_matrix -> plotting scatter matrix

these libraries are used for the visualizing the data

from sklearn import preprocessing

from sklearn.preprocessing import MinMaxScaler

these both libraries for the pre processing the data, ie., normalization and standardization

from datetime import datetime

used for the making string date to datetime format.

from sklearn.linear\_model import LinearRegression

getting Linear Regression Model.

1. **Supervised learning algorithm**

the algorithm is chosen for the problem is the linear regression

applying on the Bharti Airtel dataset for Explaining.

**Linear Regression:**

Linear regression analysis is used to predict the value of a variable based on the value of another variable.

The variable we want to predict is called the dependent variable.(Close, High)

The variable you are using to predict the other variable's value is called the independent variable. (Volume, Open)

This form of analysis estimates the coefficients of the linear equation, involving one or more independent variables that best predict the value of the dependent variable. Linear regression fits a straight line or surface that minimizes the discrepancies between predicted and actual output values. There are simple linear regression calculators that use a “least squares” method to discover the best-fit line for a set of paired data. then estimate the value of X (dependent variable) from Y (independent variable).

Here I am trying to predict the Close Value & High Value.

**Breaking Data set training and testing**

#data split 70:30

n = len(new\_data)

sev = int(0.7 \* n)

train = new\_data[:sev]

test = new\_data[sev:]

pred = ['Close','High']

x\_train = train.drop(pred, axis=1)

y\_train = train[pred]

x\_test = test.drop(pred, axis=1)

y\_test = test[pred]

**implementing Linear Regression**

**#implement linear regression**

**model = LinearRegression()**

**model.fit(x\_train,y\_train)**

**#calculating Variance and rms score**

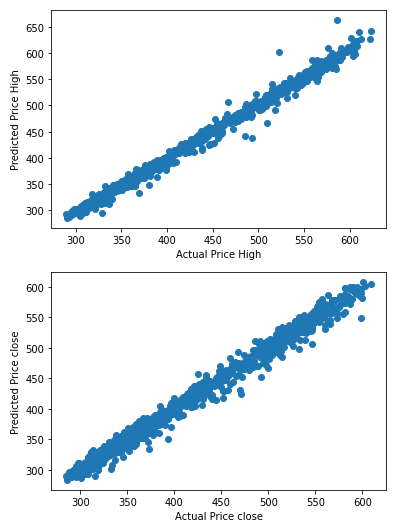
**print('Variance score: %.2f' % model.score(x\_test, y\_test))**

**rms=np.sqrt(np.mean(np.power((np.array(y\_test)-np.array(preds)),2)))**

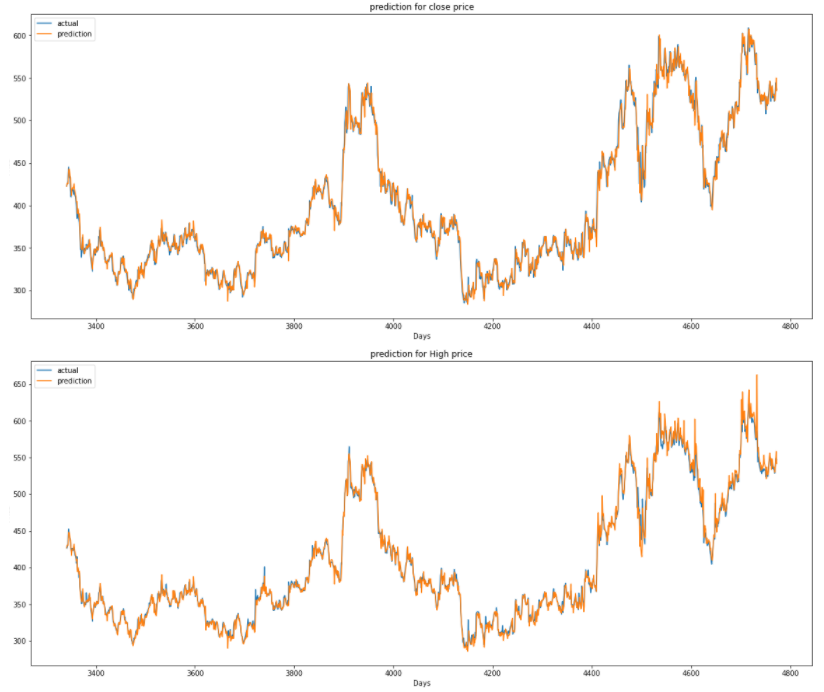
**print("rms : %.2f" %rms)**

****

**Relation B/w predicted & High**

****

**Graph prediction for the Close and High along Actual value**

****

**#accuracy of close and high**

**predsss = ['Predictions close', 'Predictions high' ]**

**y2 = test[predsss].mean()**

**x2 = test[pred].mean()**

**print('Close accuracy: %.2f' % (x2[0]/y2[0] \* 100))**

**print('High accuracy: %.2f' % (x2[1]/y2[1] \* 100))**

on implementing the model on the dataset Bharti Airtel i getting the

Close accuracy: 99.97

High accuracy: 99.94