



=

Machine Learning internship

Introduction

Instructions

Select an Industry

Module 1

Module 2

Module 3



Module 4



Module 5



Module 6



Interview



My Jupyter Notebook

Module 2

Data visualization and Technical Analysis

'A picture speaks a thousand words' has never been truer in financial markets. Absolutely no one goes through the millions of rows of numbers, we always prefer the data in a plotted form to draw better inferences. This module would cover the plotting, basic technical indicators and our own customisation, and making our own trade calls!

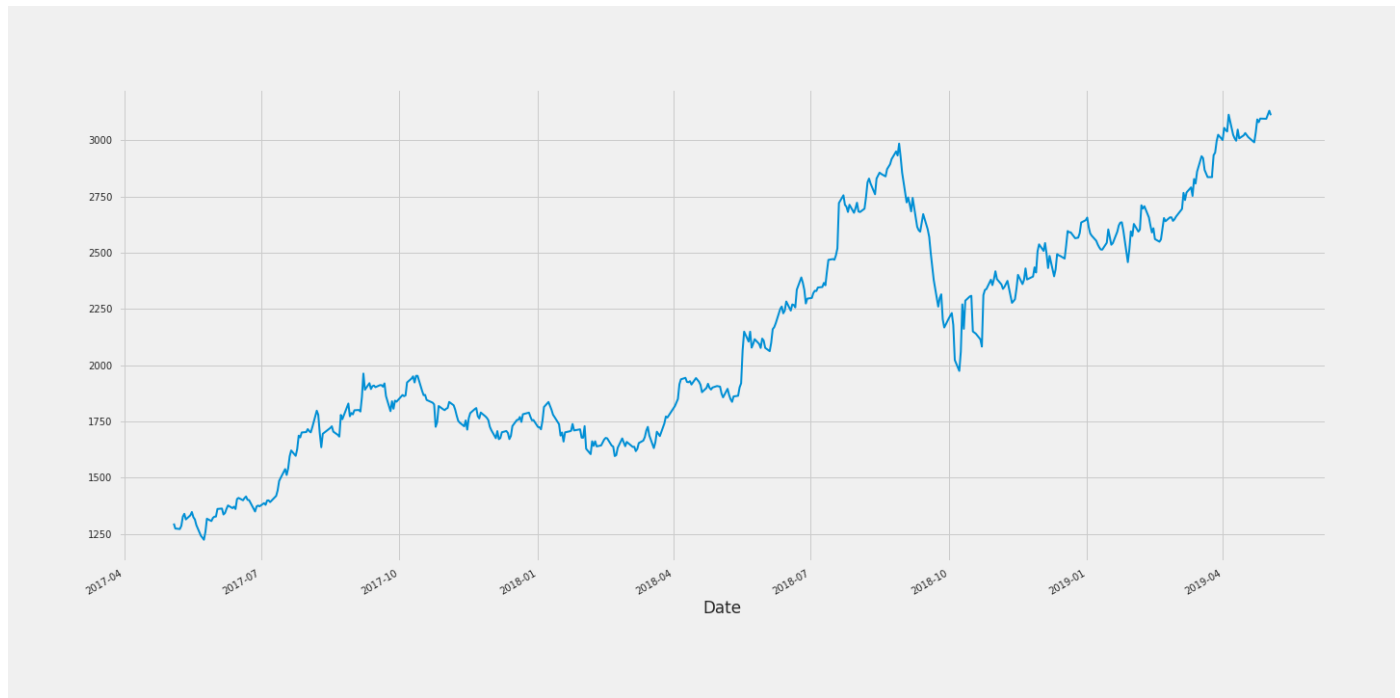
You should target to finish module 2, including the prerequisites, in 1-2 weeks.

Problem Statements

2.1 Load the week2.csv file into a dataframe. What is the type of the Date column? Make sure it is of type datetime64. Convert the Date column to the index of the dataframe.

Plot the closing price of each of the days for the entire time frame to get an idea of what the general outlook of the stock is.

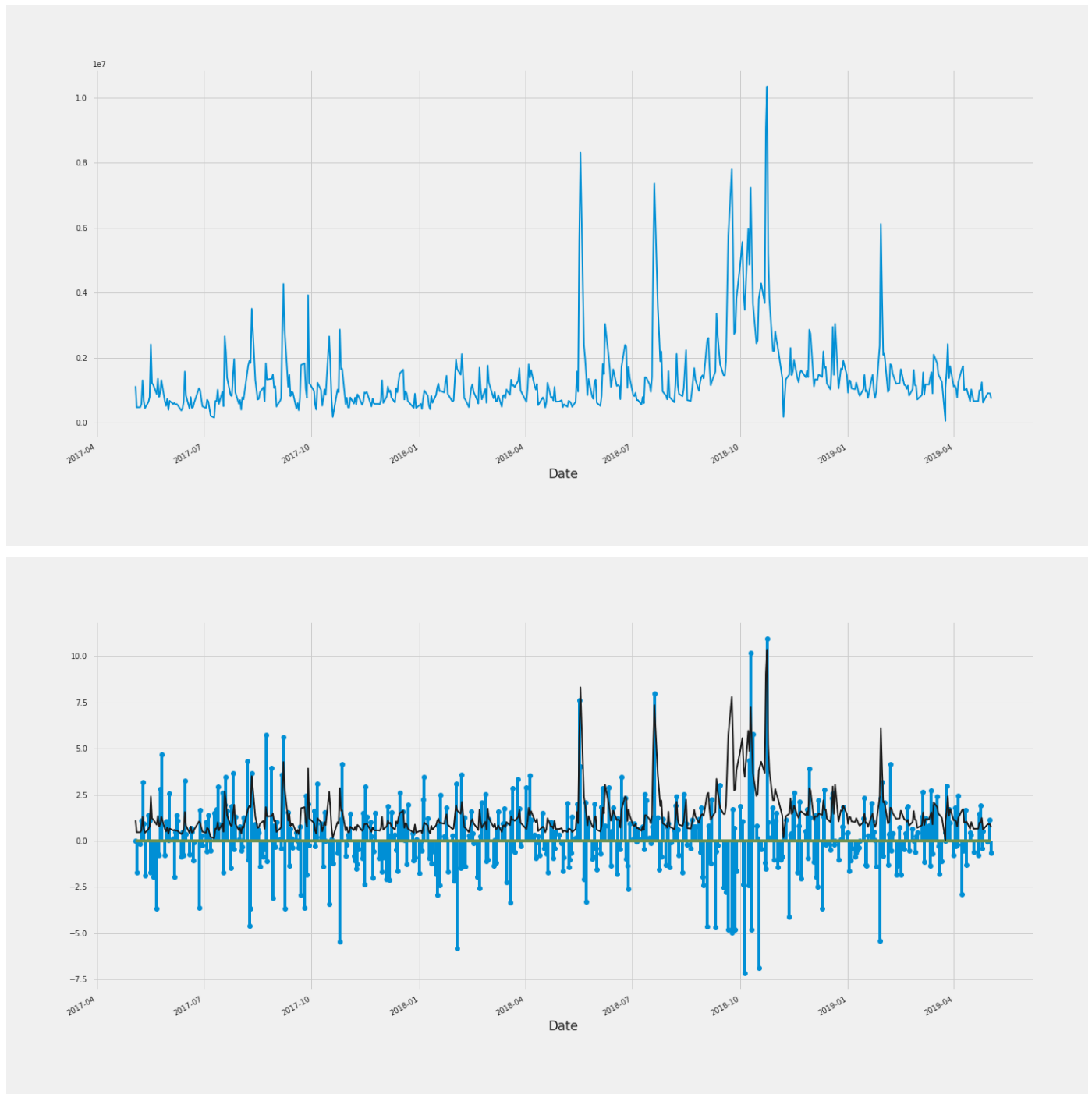
- ✓ Look out for drastic changes in this stock, you have the exact date when these took place, try to fetch the news for this day of this stock
- ✓ This would be helpful if we are to train our model to take NLP inputs.



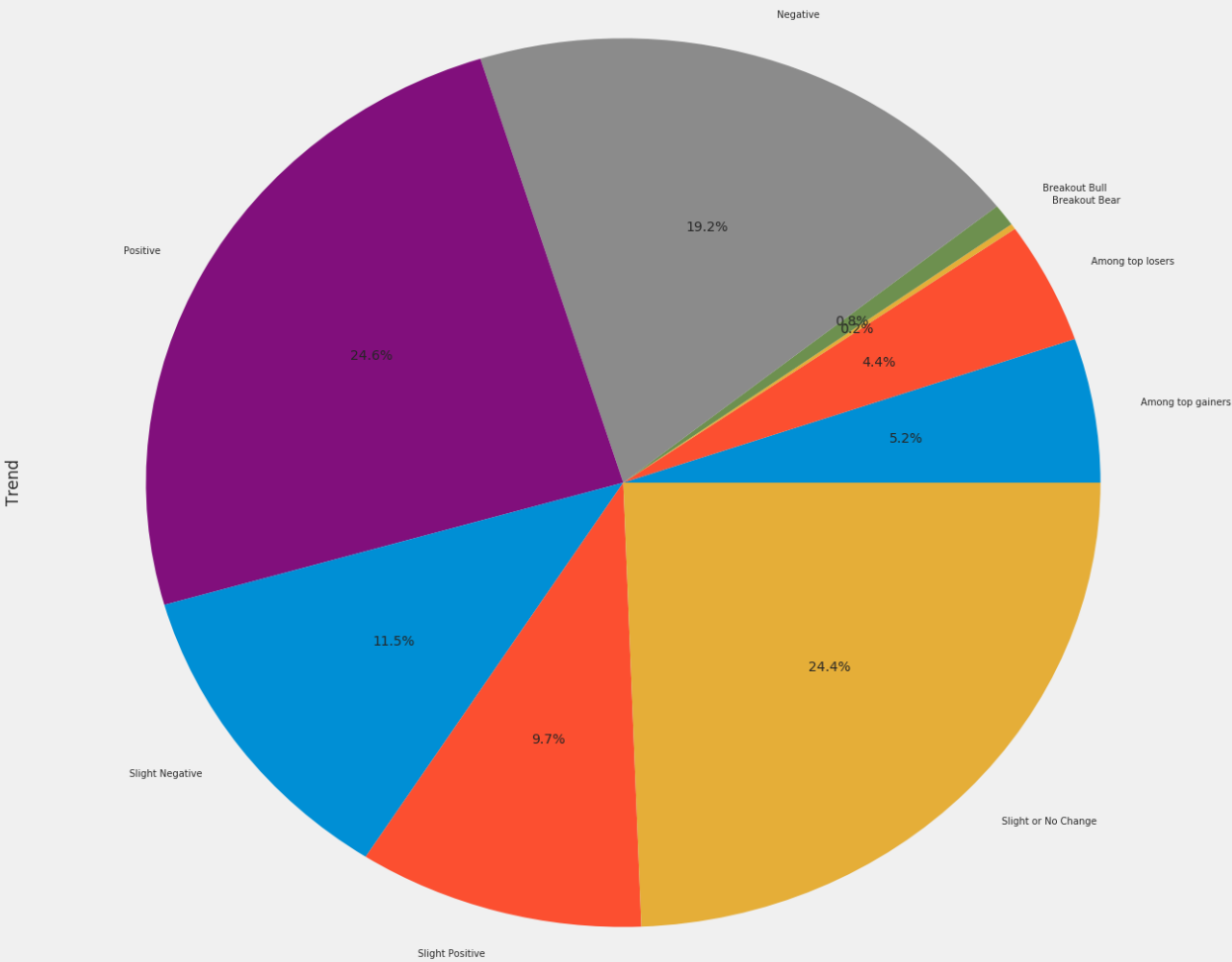
2.2 A stem plot is a discrete series plot, ideal for plotting daywise data. It can be plotted using the `plt.stem()` function.

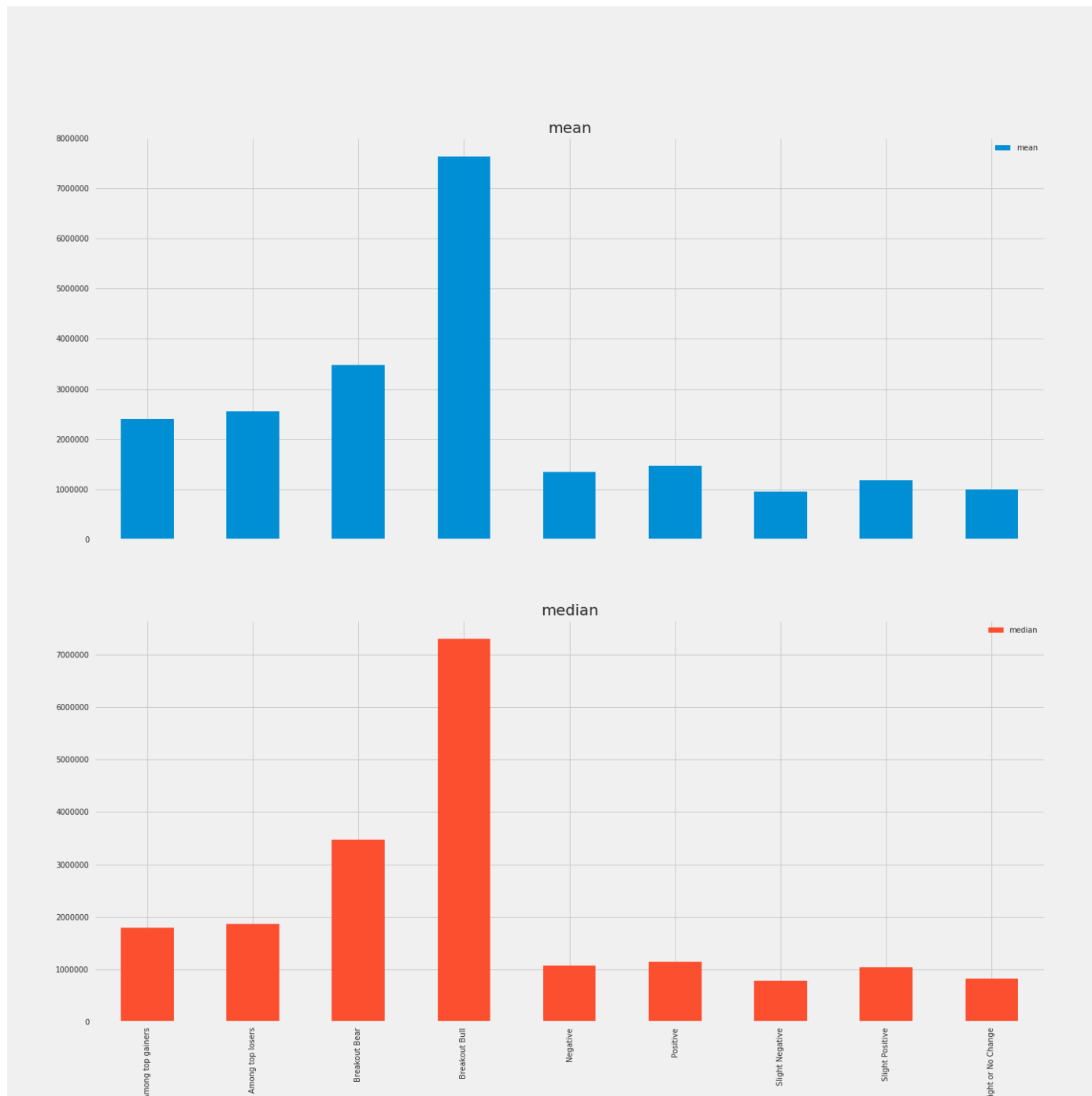
Display a stem plot of the daily change in of the stock price in percentage. This column was calculated in module 1 and should be already available in `week2.csv`. Observe whenever there's a large change.

2.3 Plot the daily volumes as well and compare the percentage stem plot to it. Document your analysis of the relationship between volume and daily percentage change.



2.4 We had created a Trend column in module 1. We want to see how often each Trend type occurs. This can be seen as a pie chart, with each sector representing the percentage of days each trend occurs. Plot a pie chart for all the 'Trend' to know about relative frequency of each trend. You can use the groupby function with the trend column to group all days with the same trend into a single group before plotting the pie chart. From the grouped data, create a BAR plot of average & median values of the 'Total Traded Quantity' by Trend type.





2.5 Plot the daily return (percentage) distribution as a histogram.

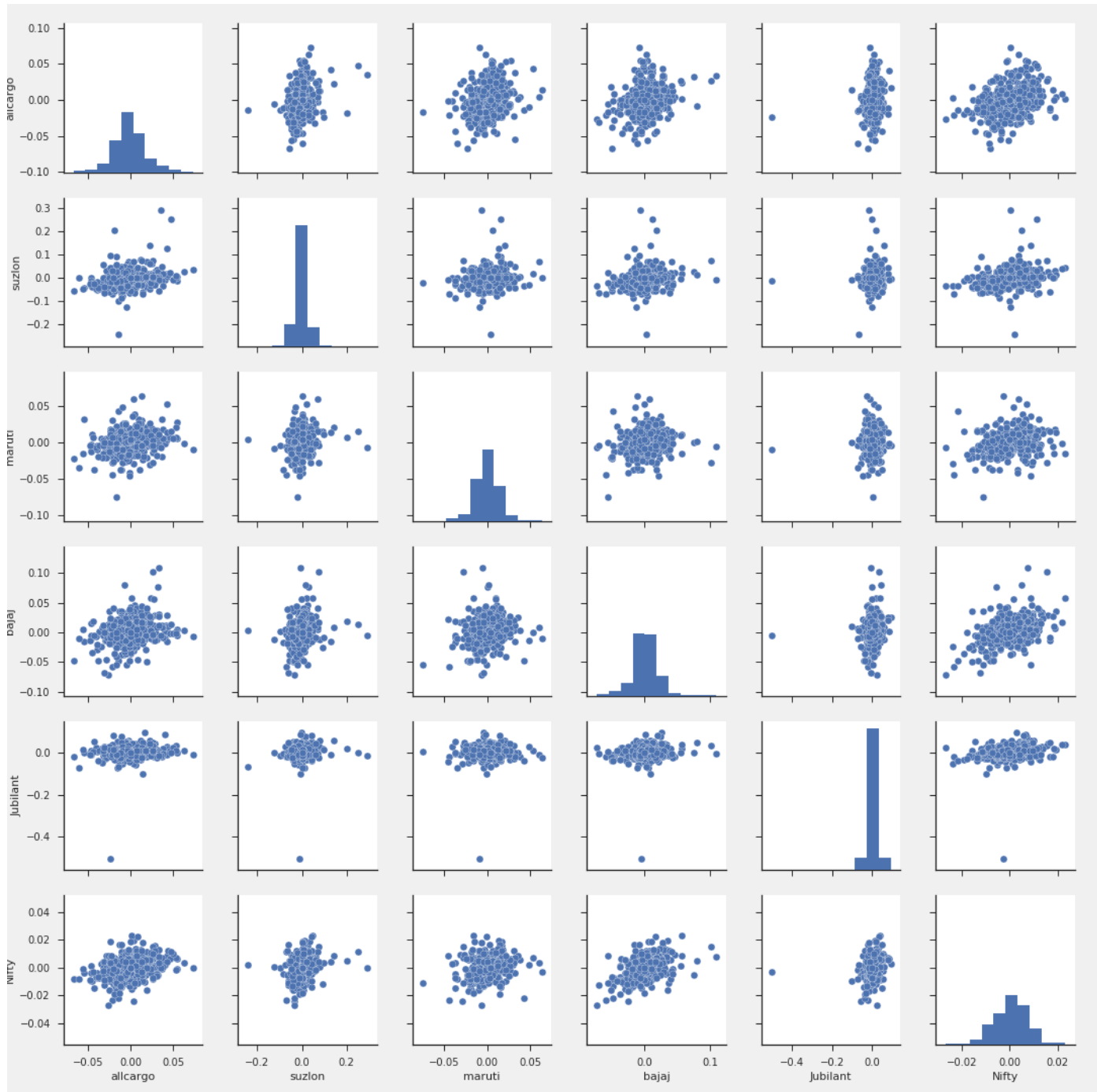
Histogram analysis is one of the most fundamental methods of exploratory data analysis. In this case, it'd return a frequency plot of various values of percentage changes .

2.6 We next want to analyse how the behaviour of different stocks are correlated. The correlation is performed on the percentage change of the stock price instead of the stock price.

Load any 5 stocks of your choice into 5 dataframes. Retain only rows for which 'Series' column has value 'EQ'. Create a single dataframe which contains the 'Closing Price' of each stock. This dataframe should hence have five columns. Rename each column to the name of the stock that is contained in the column. Create a new dataframe which is a percentage change of the values in the previous dataframe. Drop Nan's from this dataframe.

Using seaborn, analyse the correlation between the percentage changes in the five stocks. This is extremely useful for a fund manager to design a diversified portfolio. To know more, check out these resources on correlation (<https://www.investopedia.com/terms/c/correlation.asp>) and diversification

(<https://www.investopedia.com/terms/d/diversification.asp>).

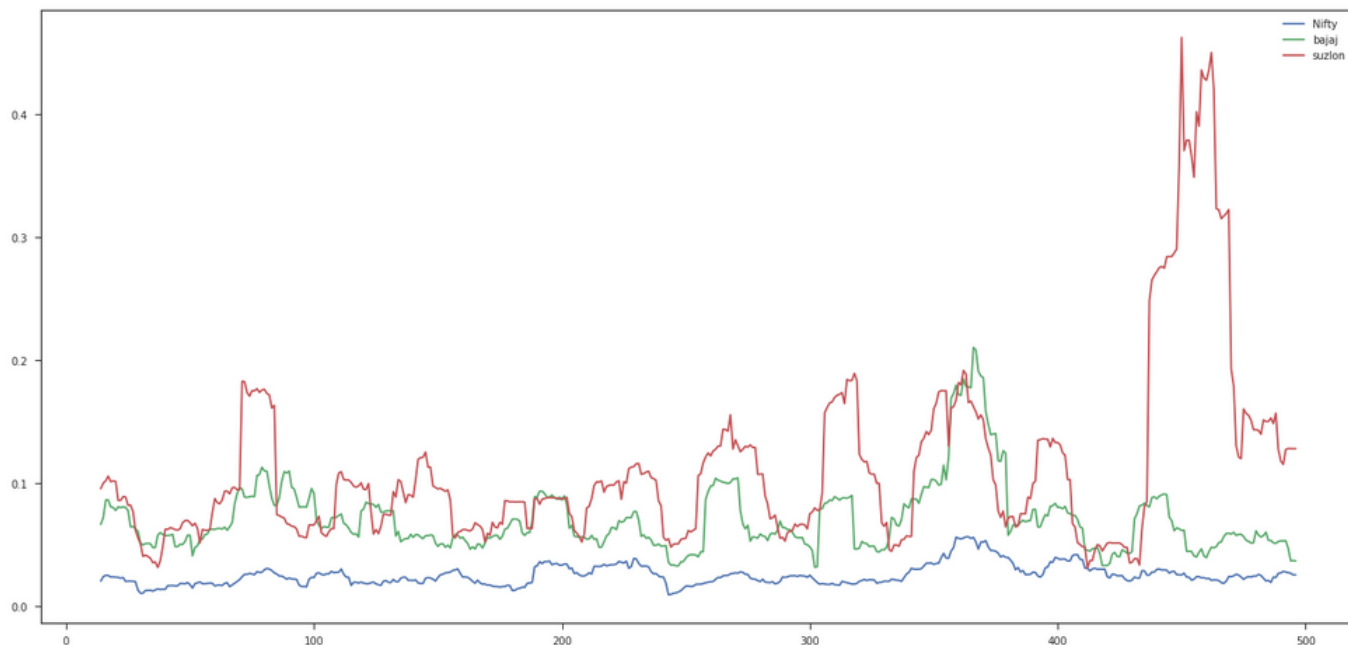


2.7 Volatility is the change in variance in the returns of a stock over a specific period of time. Do give the following documentation on volatility (<https://www.investopedia.com/terms/v/volatility.asp>) a read.

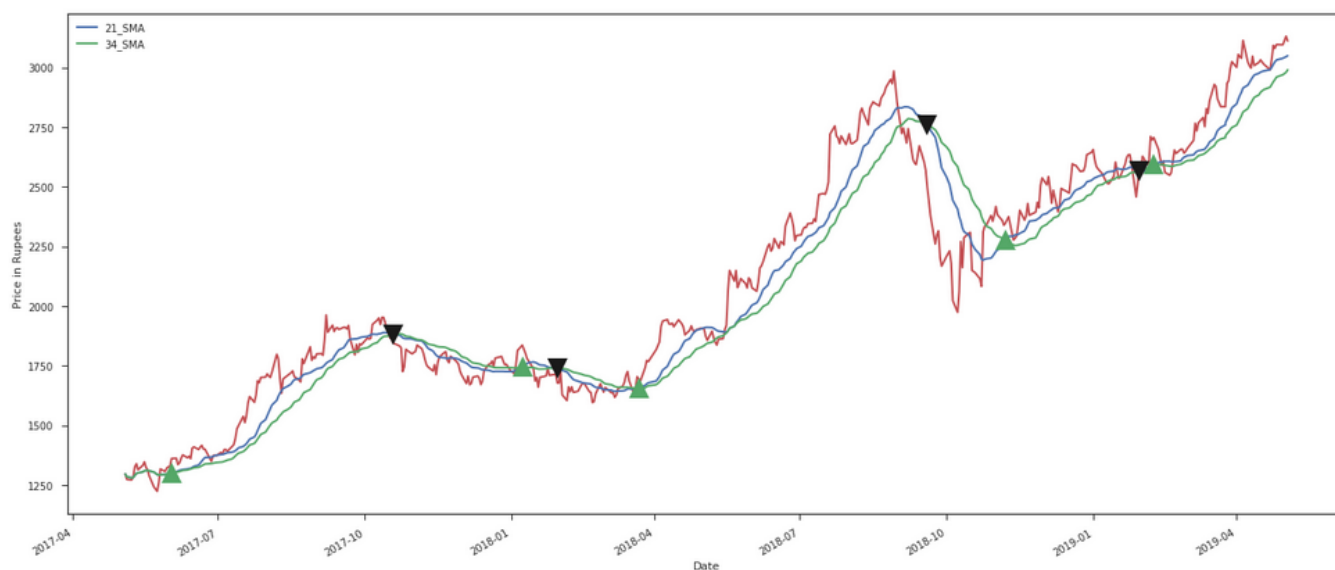
You have already calculated the percentage changes in several stock prices. Calculate the 7 day rolling average of the percentage change of any of the stock prices, then compute the standard deviation (which is the square root of the variance) and plot the values.

Note: pandas provides a `rolling()` function for dataframes and a `std()` function also which you can use.

2.8 Calculate the volatility for the Nifty index and compare the 2. This leads us to a useful indicator known as 'Beta' (We'll be covering this in length in Module 3)



2.9 Trade Calls - Using Simple Moving Averages. Study about moving averages here (<https://www.investopedia.com/terms/s/sma.asp>).



Plot the 21 day and 34 day Moving average with the average price and decide a Call !

Call should be buy whenever the smaller moving average (21) crosses over longer moving average (34) AND the call should be sell whenever smaller moving average crosses under longer moving average.

One of the most widely used technical indicators.

2.10 Trade Calls - Using Bollinger Bands (<https://www.investopedia.com/terms/b/bollingerbands.asp>)

Plot the bollinger bands for this stock - the duration of 14 days and 2 standard deviations away from the average

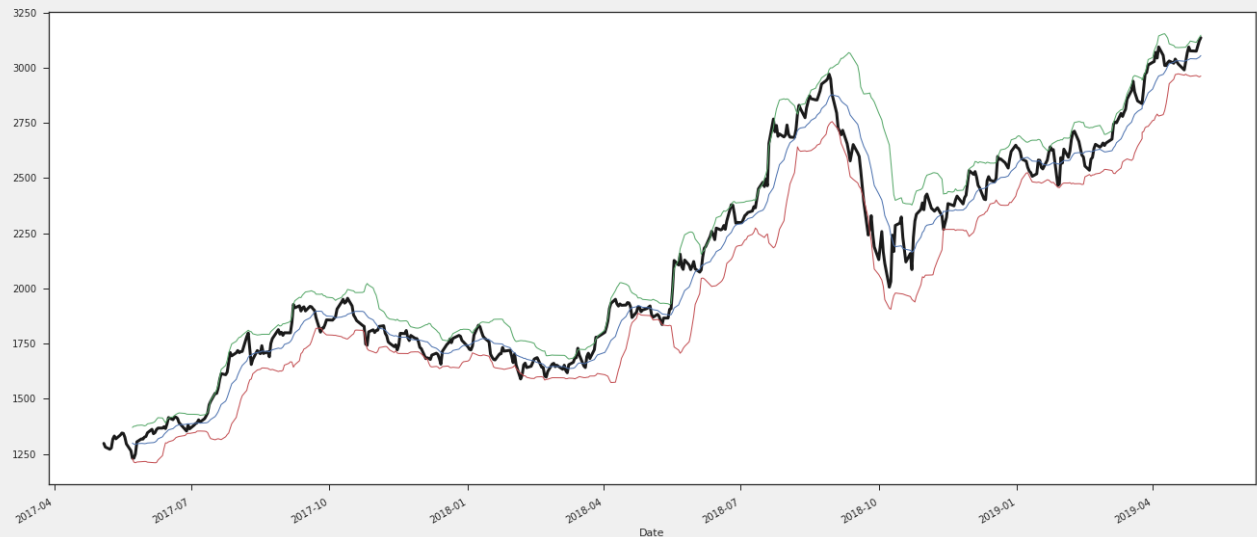
The bollinger bands comprise the following data points-

- ✓ The 14 day rolling mean of the closing price (we call it the average)
- ✓ Upper band which is the rolling mean + 2 standard deviations away from the average.
- ✓ Lower band which is the rolling mean - 2 standard deviations away from the average.
- ✓ Average Daily stock price.

Bollinger bands are extremely reliable , with a 95% accuracy at 2 standard deviations , and especially useful in sideways moving market.

Observe the bands yourself , and analyse the accuracy of all the trade signals provided by the bollinger bands.

Save to a new csv file.



Submit

Getting stuck and need a little guidance? Don't fret cause help is on the way! Take a look at our [guidance video ► \(\)](#) on the topic.

aspiration.ai (/)

[About Us \(/about.jsp\)](#) [Contact Us \(/contact-us.jsp\)](#) [Help \(/support.jsp\)](#)

© 2018 CL Educate Ltd. All rights reserved.



(//www.facebook.com/CLEducate)