IME 611 TASK A REPORT
GROUP 9

AYUSHI GOYAL	SHUBHAM GUPTA	SAMPADA SINHA
180182	180749	180658

No member in our group has done any course on Time Series.

We first use pandas DataReader to get the stock price data for both Nifty 50 and Nifty 100 stocks. We found that many companies in these indices were not listed in the period provided to us, so we dropped them from our initial dataset; there were two such stocks.
We will primarily focus on the Nifty 50 universe for this project. We are also reducing the time period for our studies in some cases due to COVID19, as it had a devastating effect on the market as we know it.

TASK 1:
(CODE: https://bit.ly/3fSlhKK)

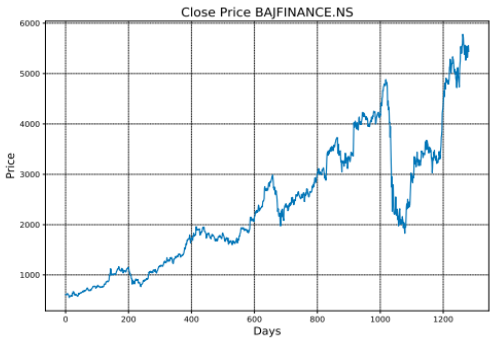
In every time series, there is level(mean of the series) and noise(random variation in that series). Trends and seasonality are optional in a time series. Since we know that stock prices are influenced by a lot of internal and external factors, we are interested to see if some of these stocks have a trend or a seasonality. If we could find one, we can optimize our trading algorithms in a way to increase our profitability and lower the risks.
A time series can be thought of as either an addition or a multiplication of level, noise, trend, and seasonality.
Decomposition is the deconstruction of the series data into its various components: trend, cycle, noise, and seasonality when those exist. We will use it to find components in both models (additive and multiplicative).
In the hope of getting some trend and seasonality, we analyzed data till Jan 2020, before the COVID pandemic.

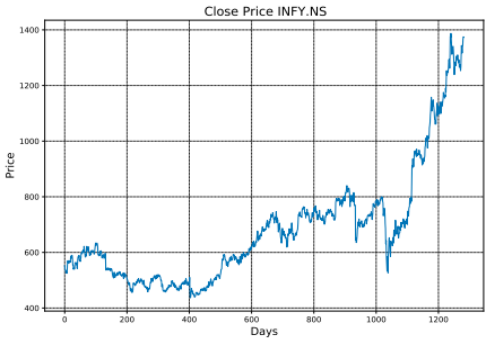
We have shown graphs of daily price variation, log return, and its time series characteristics followed by returns and its time series characteristics for six stocks from different sectors, like Banking, Finance, FMCG, Pharmaceutical, IT and Automobile.

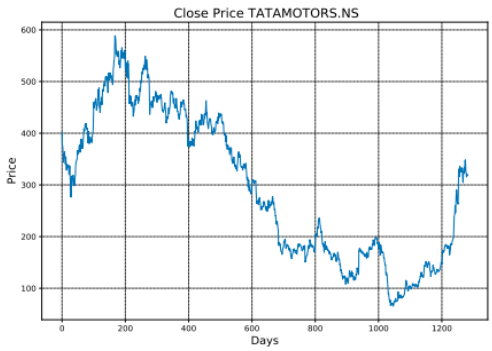
The graphs for all others can be found in the link:
Price: https://bit.ly/2PIFIV8
Time Series Characteristics for Prices: https://bit.ly/3uw6Bot

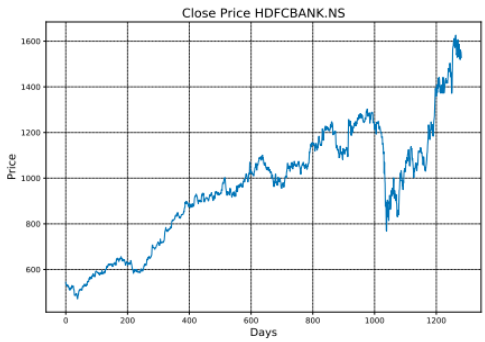
Log Return: https://bit.ly/3rVKtSN
Time Series Characteristics for Log Return: https://bit.ly/3fRgauk
Return: https://bit.ly/2RhohpJ
Time Series Characteristics for Return: https://bit.ly/3fQeWzt

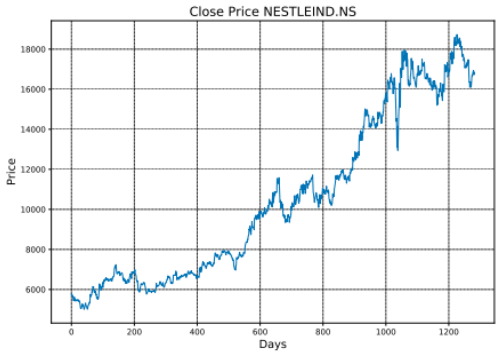
PRICES:



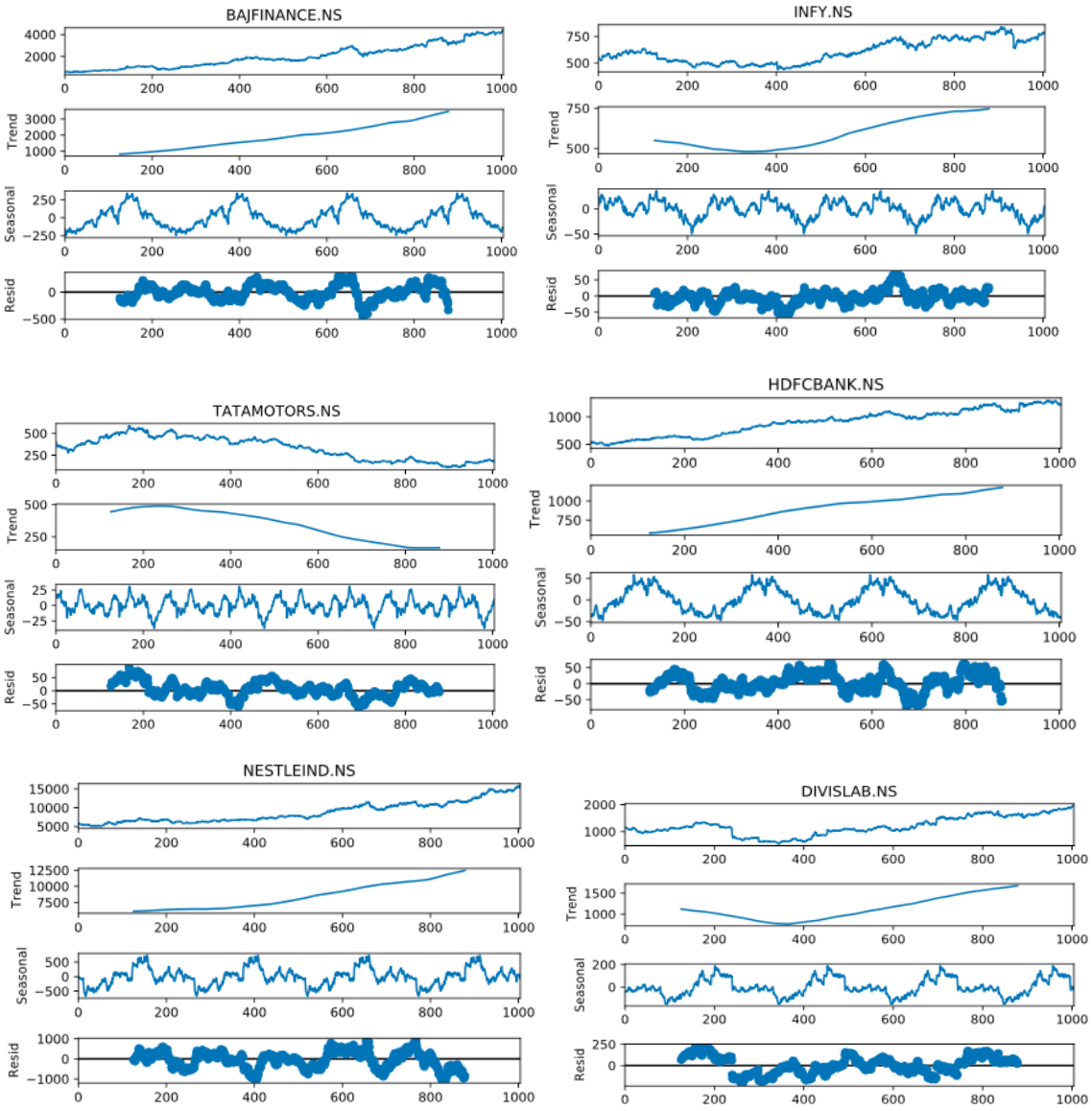




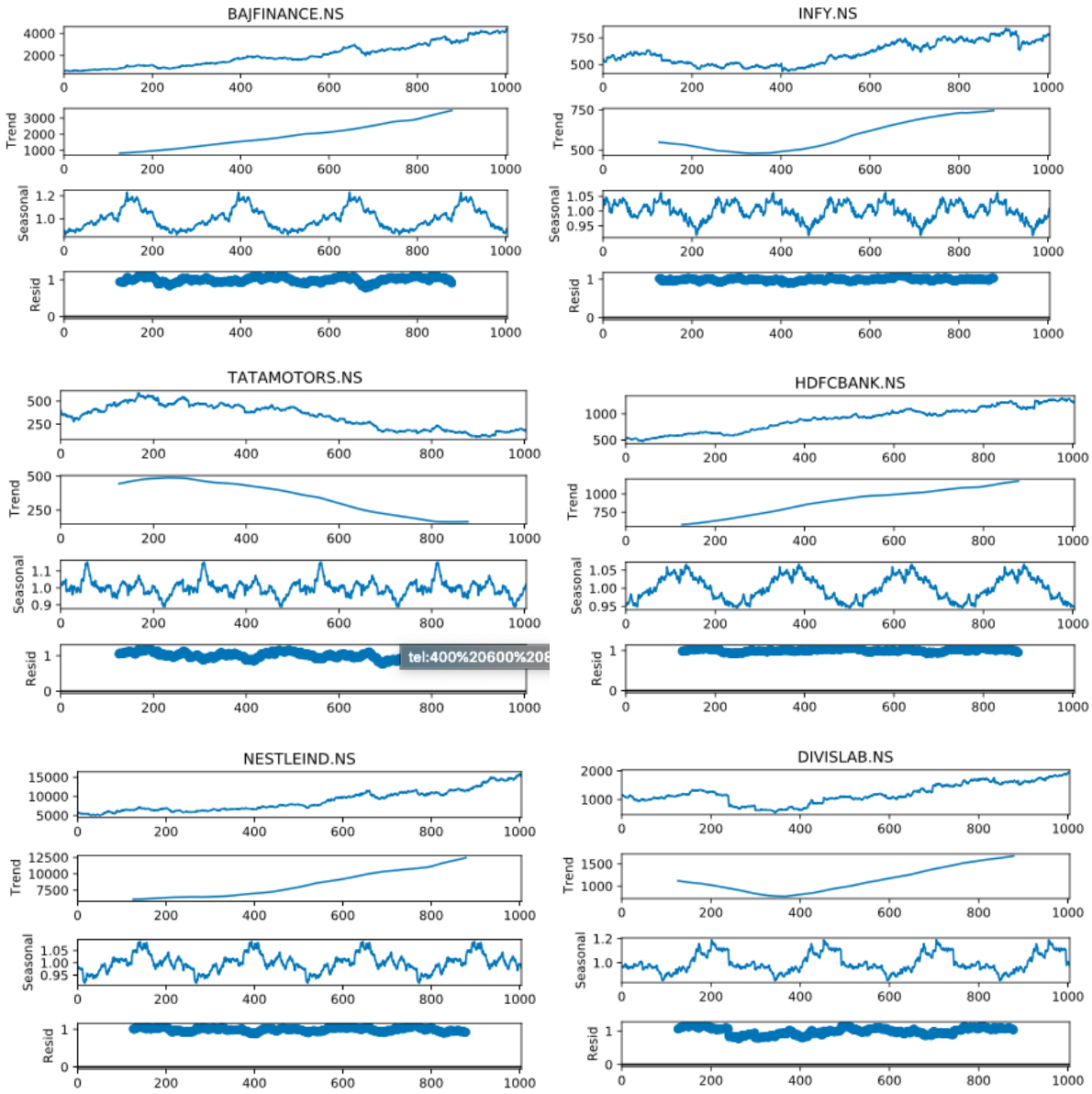




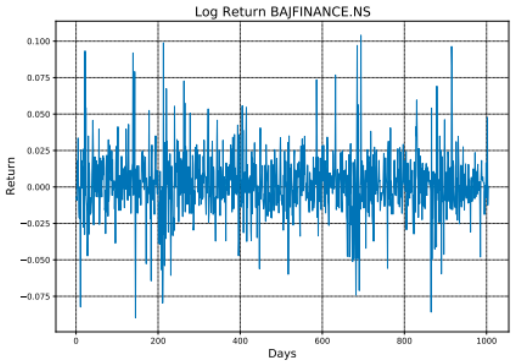
TIME SERIES FOR DAILY PRICES (ADDITIVE):

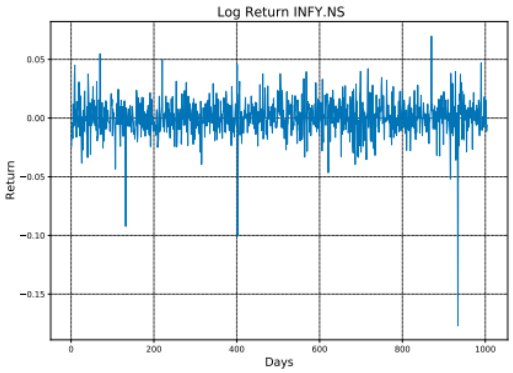


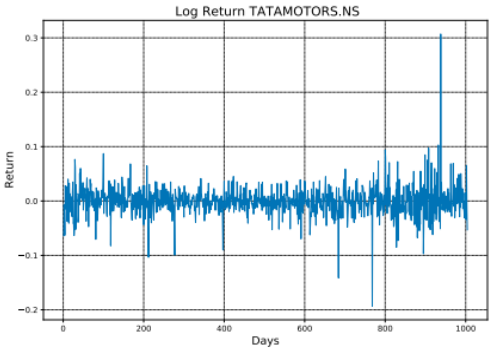
TIME SERIES FOR DAILY PRICES (MULTIPLICATIVE):

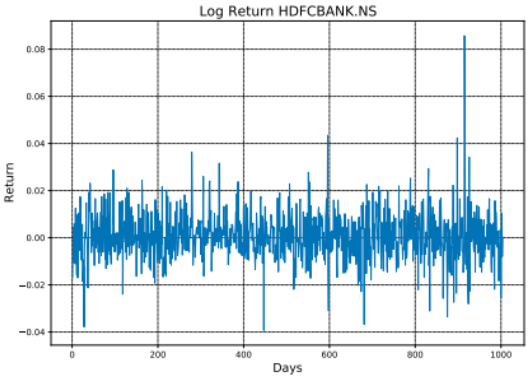


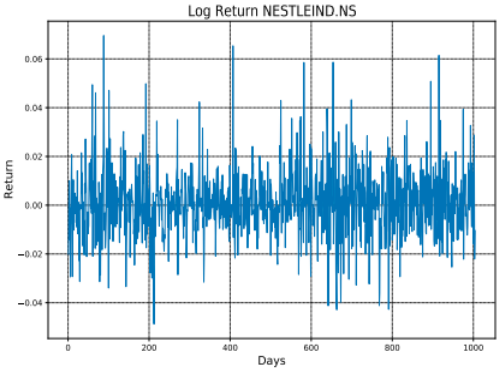
LOG RETURN:

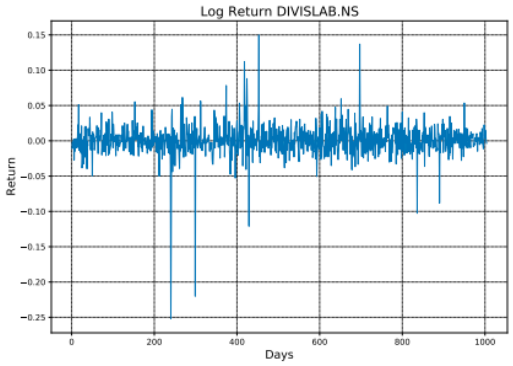

Log Return BAJFINANCE.NS
Return
Days


Log Return INFY.NS
Return
Days

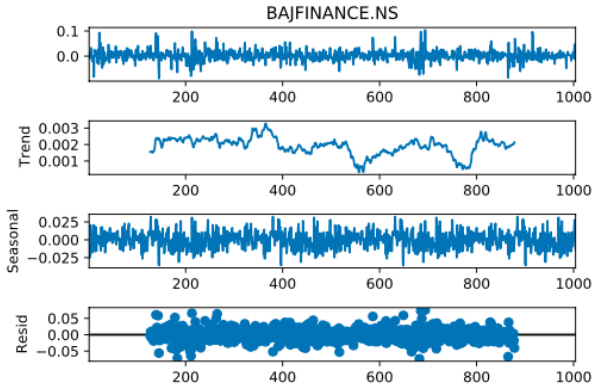

Log Return TATAMOTORS.NS
Return
Days

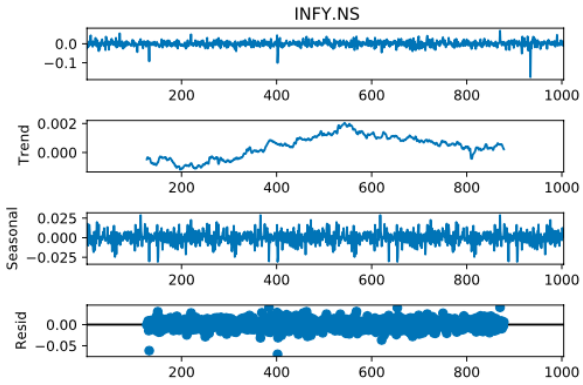

Log Return HDFCBANK.NS
Return
Days

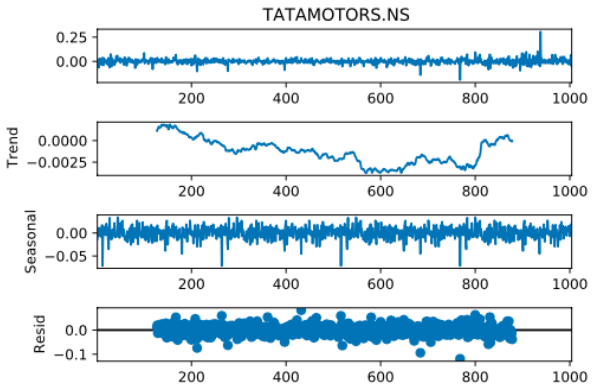

Log Return NESTLEIND.NS
Return
Days

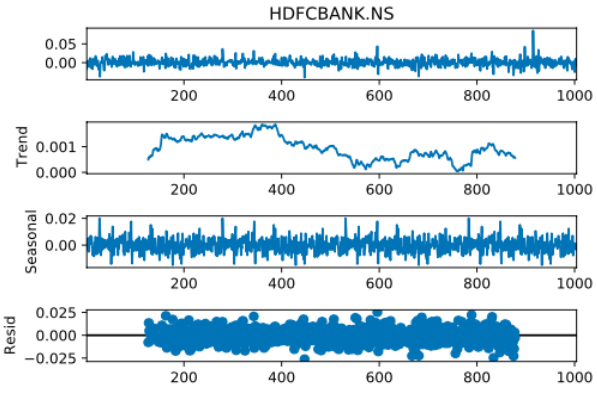

Log Return DIVISLAB.NS
Return
Days

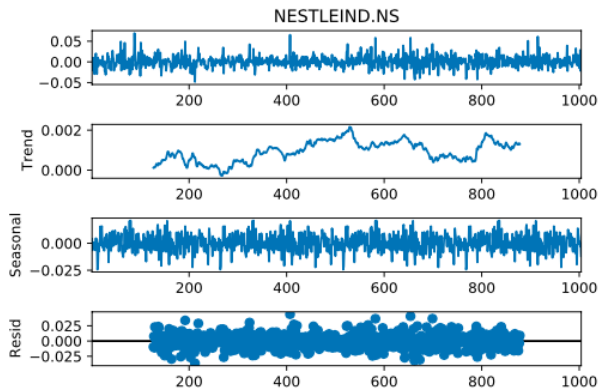
TIME SERIES FOR LOG RETURN (ADDITIVE ONLY):
MULTIPLICATIVE DOES NOT EXIST FOR NEGATIVE VALUES

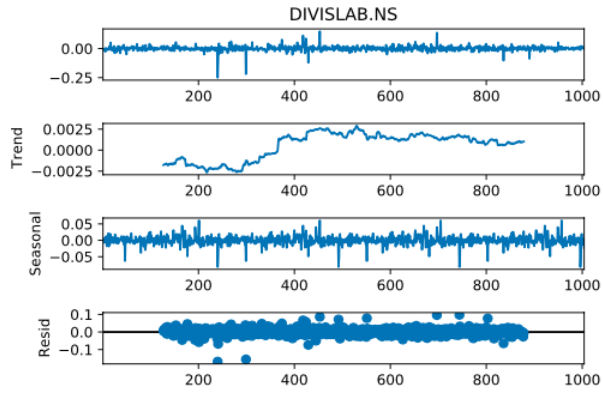




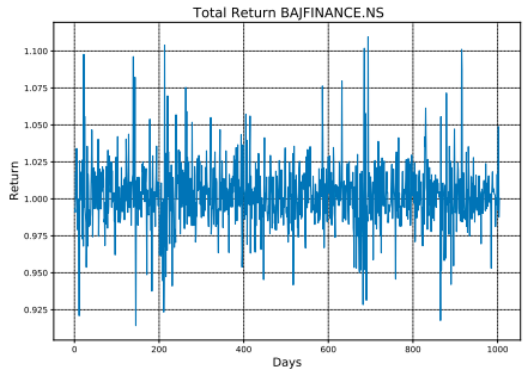


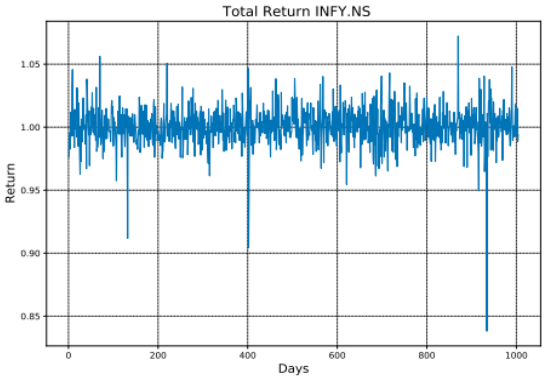


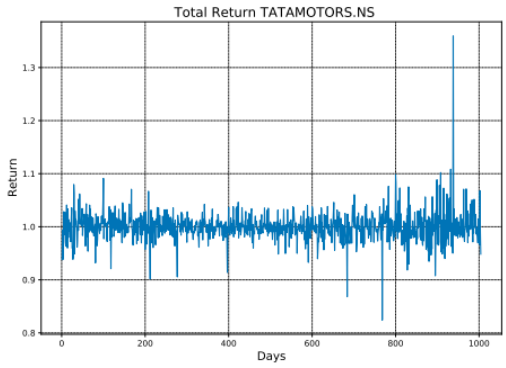


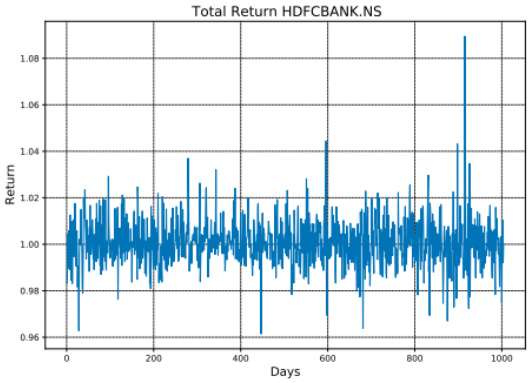


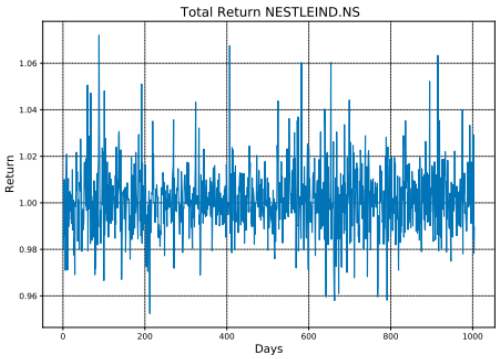
RETURN:

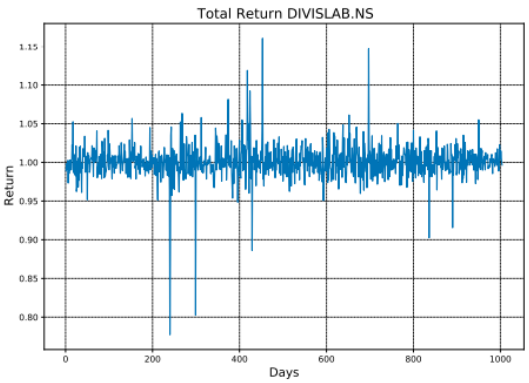

Total Return BAJFINANCE.NS
Return
Days


Total Return INFY.NS
Return
Days

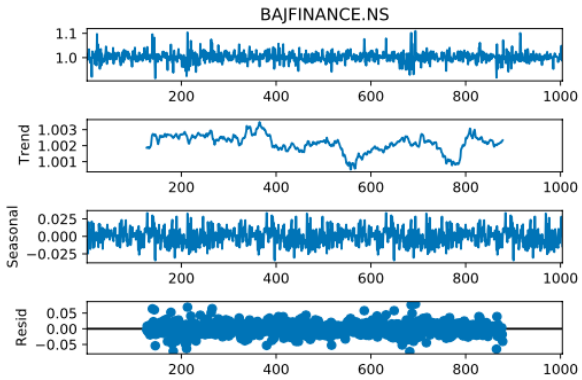

Total Return TATAMOTORS.NS
Return
Days

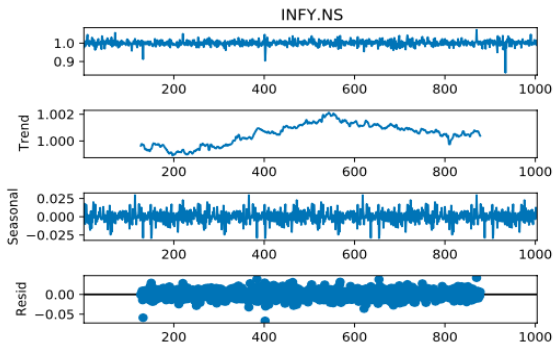

Total Return HDFCBANK.NS
Return
Days

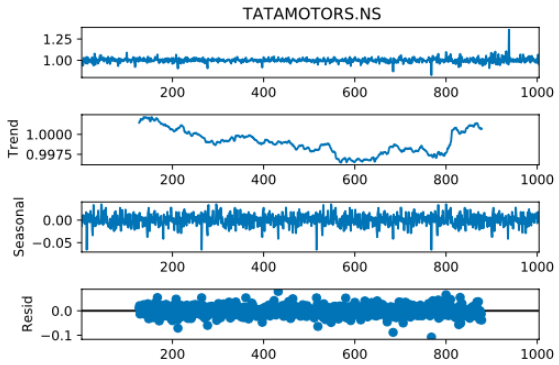

Total Return NESTLEIND.NS
Return
Days

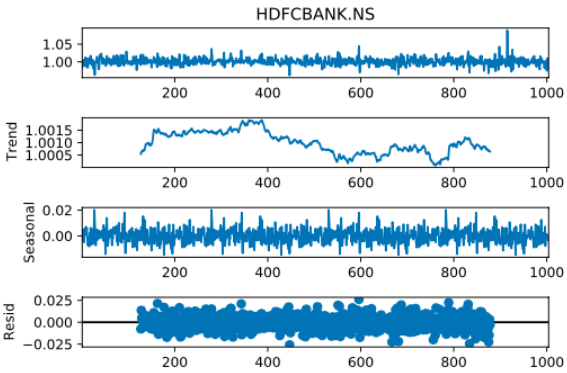

Total Return DIVISLAB.NS
Return
Days

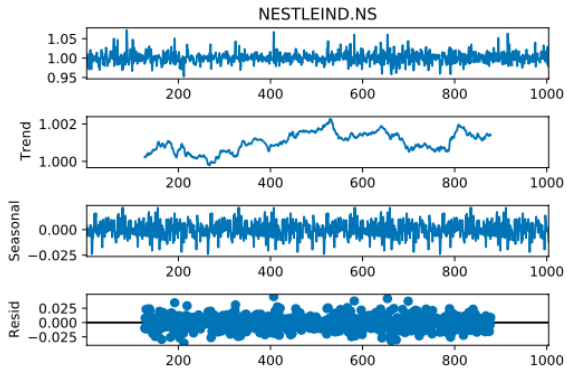
TIME SERIES FOR RETURNS (ADDITIVE):

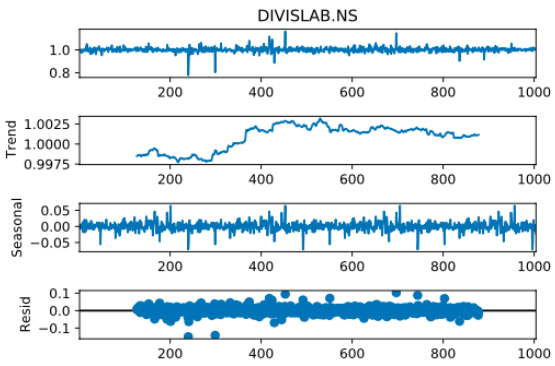




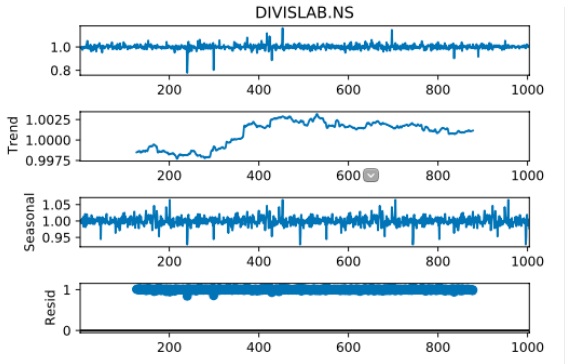


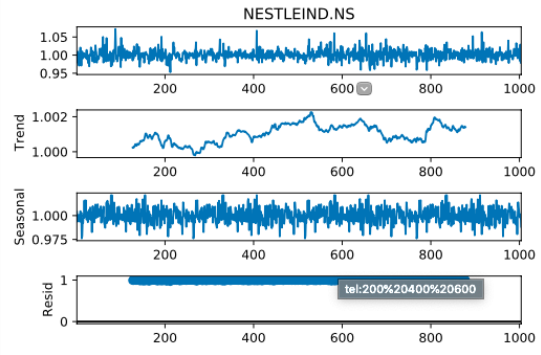


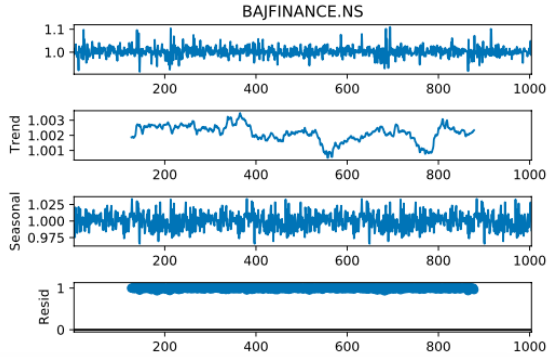


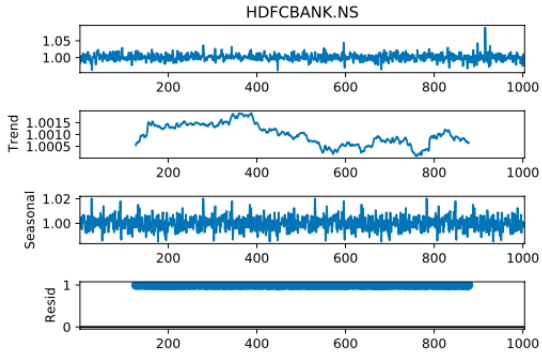


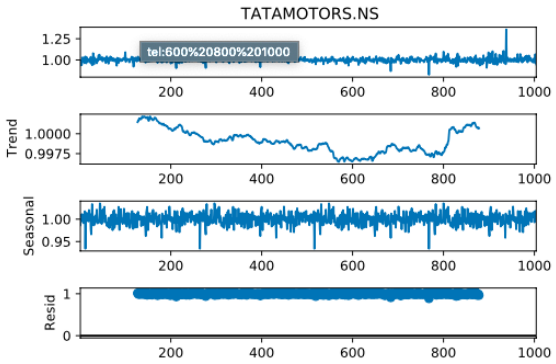
TIME SERIES FOR RETURNS (MULTIPLICATIVE):

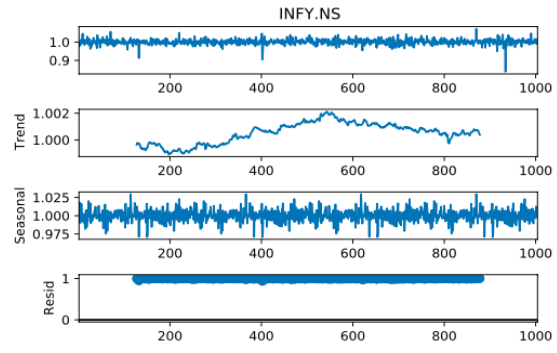












TASK 2:
(CODE: https://bit.ly/3msnFJ8)

Determining normal distribution or not?

For this task, we have used the Shapiro Wilk Test, and we have identified that when returns are calculated at an interval of 5 days, most of the distributions are normal or nearly normal. The Shapiro Wilk Test is the most powerful test, and the interpretation is as follows:
P_Value > 0.05 then, assume normal distribution
P_Value < 0.05 then, not normal distribution

P_Value according to the Shapiro Test for each stock:

Sr. no.	Stock name	p-value (Shapiro Wilk test)	Is distribution normal?
1	ADANIPORTS.NS	0.004577595274895 43	No
2	TECHM.NS	1.190663442685036e-05	No
3	BRITANNIA.NS	0.06534425914287567	Yes
4	TCS.NS	0.41729554533958435	Yes
5	SUNPHARMA.NS	3.9420359826181084e-05	No
6	INFY.NS	0.026289535686373 71	No
7	HINDALCO.NS	0.21124500036239624	Yes
8	DR REDDY.NS	2.1592366783451666e-10	No
9	HCLTECH.NS	5.519610226656368e-07	No
10	HINDUNILVR.NS	0.42400506138801575	Yes
11	TATASTEEL.NS	0.20954789221286774	Yes

12	NTPC.NS	1.0743862731033005e-05	No
13	CIPLA.NS	0.529323399066925	Yes
14	ITC.NS	4.585757778841071e-05	No
15	BPCL.NS	0.46152183413505554	Yes
16	WIPRO.NS	0.0013912224676460028	No
17	JSWSTEEL.NS	0.012935689650475979	No
18	GAIL.NS	0.009982343763113022	Yes
19	DIVISLAB.NS	0.2680056393146515	Yes
20	ULTRACEMCO.NS	0.24597375094890594	No
21	NESTLEIND.NS	0.016239793971180916	Yes
22	ASIANPAINT.NS	0.4691813886165619	No
23	EICHERMOT.NS	7.962701783981174e-05	No
24	IOC.NS	0.011599199846386 91	No
25	GRASIM.NS	5.127526181425424e-10	No
26	BAJAJ-AUTO.NS	0.2190645933151245	Yes
27	HDFC.NS	0.7684739232063293	Yes
28	M&M.NS	0.17216472327709198	Yes
29	UPL.NS	0.011371889151632786	No
30	SHREECEM.NS	0.18403418362140656	Yes
31	MARUTI.NS	0.0005735829472541809	No
32	TITAN.NS	0.2779155969619751	Yes
33	BAJAJFINSV.NS	0.2487795203924179	Yes
34	KOTAKBANK.NS	0.016460537910461426	No
35	COALINDIA.NS	0.0007262146100401878	No

36	BHARTIARTL.NS	0.21439394354820251	Yes
37	HEROMOTOCO.NS	0.025956494733691216	No
38	LT.NS	3.2027152041536056e-09	No
39	ONGC.NS	0.009697488509118557	No
40	SBIN.NS	5.1080507546430454e-05	No
41	RELIANCE.NS	0.16058485209941864	Yes
42	BAJFINANCE.NS	0.006546803750097752	No
43	AXISBANK.NS	0.207569420337677	Yes
44	HDFCBANK.NS	0.0001880933268694207	No
45	TATAMOTORS.NS	0.17804868519306183	Yes
46	ICICIBANK.NS	0.5358592867851257	Yes
47	POWERGRID.NS	0.0032723774202167988	No
48	INDUSINDBK.NS	2.3507946025347337e-05	No

Now we performed a chi-square test for each stock to check whether estimating them as normal is suitable or not. The chi square statistic x^2 is calculated by the following steps:

1. For each *observed* number in the table subtract the corresponding *expected* number (*O* — *E*).
2. Square the difference [$(O - E)^2$].
3. Divide the squares obtained for each cell in the table by the *expected* number for that cell [$(O - E)^2 / E$].
4. Sum all the values for $(O - E)^2 / E$. This is the chi square statistic.

The least value of chi-square tells us the most predicted distribution.
We observed that for all the stocks normal ranks in the top 4 of the best predicted distribution. As the value of chi-square ranges up to 20000, a difference of 10 or 20 can be approximated as it is almost 0.05-0.1% of it. Thus even if normal lies in the top 4, we can approximate the distributions as normal.

Some snippets of stocks and their chi-square values are shown below which supports the above statement.

	Distribution	chi_square
3	beta	25.581285
1	norm	34.866762
8	lognorm	35.888111
0	weibull_min	35.974494
2	weibull_max	43.880193
6	gamma	46.685485
4	invgauss	89.181302
10	triang	263.501372
5	uniform	1397.472679
7	expon	3024.983281
9	pearson3	20453.660116

	Distribution	chi_square
1	norm	33.022178
9	pearson3	37.851713
6	gamma	37.853671
8	lognorm	38.104940
4	invgauss	40.391679
0	weibull_min	61.240310
2	weibull_max	81.862725
10	triang	138.221633
5	uniform	614.874396
7	expon	2409.926144
3	beta	4902.000000

	Distribution	chi_square
0	weibull_min	4.800461
3	beta	5.377865
2	weibull_max	5.411609
1	norm	7.881494
8	lognorm	8.245652
6	gamma	11.200023
4	invgauss	33.442911
10	triang	56.180535
5	uniform	1011.835047
7	expon	2735.330703
9	pearson3	20272.470026

	Distribution	chi_square
3	beta	25.307994
1	norm	25.324069
9	pearson3	25.350928
6	gamma	25.357774
8	lognorm	43.154678
4	invgauss	53.758915
2	weibull_max	64.535178
0	weibull_min	99.413290
10	triang	604.540842
5	uniform	1804.955985
7	expon	5084.034104

We also predicted the actual distribution of return of each stock which is as follows:

Sr. No.	**Stock**	**Distribution**
1	ADANIPORTS.NS	beta
2	TECHM.NS	Norm
3	BRITANNIA.NS	beta
4	TCS.NS	weibull_min
5	SUNPHARMA.NS	beta
6	INFY.NS	beta
7	HINDALCO.NS	Norm

8	DR REDDY.NS	beta
9	HCLTECH.NS	beta
10	HINDUNILVR.NS	beta
11	TATASTEEL.NS	Norm
12	NTPC.NS	weibull_min
13	CIPLA.NS	beta
14	ITC.NS	Norm
15	BPCL.NS	pearson3
16	WIPRO.NS	beta
17	JSWSTEEL.NS	beta
18	GAIL.NS	lognorm
19	DIVISLAB.NS	gamma
20	ULTRACEMCO.NS	pearson3
21	NESTLEIND.NS	lognorm
22	ASIANPAINT.NS	weibull_min
23	EICHERMOT.NS	beta
24	IOC.NS	lognorm
25	GRASIM.NS	beta
26	BAJAJ-AUTO.NS	Norm
27	HDFC.NS	Norm
28	M&M.NS	lognorm
29	UPL.NS	lognorm
30	SHREECEM.NS	beta
31	MARUTI.NS	beta

32	TITAN.NS	Norm
33	BAJAJFINSV.NS	lognorm
34	KOTAKBANK.NS	lognorm
35	COALINDIA.NS	beta
36	BHARTIARTL.NS	beta
37	HEROMOTOCO.NS	invgauss
38	LT.NS	lognorm
39	ONGC.NS	beta
40	SBIN.NS	Norm
41	RELIANCE.NS	weibull_min
42	BAJFINANCE.NS	Norm
43	AXISBANK.NS	weibull_min
44	HDFCBANK.NS	beta
45	TATAMOTORS.NS	lognorm
46	ICICIBANK.NS	Norm
47	POWERGRID.NS	Norm
48	INDUSINDBK.NS	beta

TASK 3:
(CODE: https://bit.ly/3fPwifA)

For the calculation of expected return we have first calculated the annual return for each year and then reported the average of those values over 5 years. Here is the annual return of each stock for five years

S. No.	Stock	2016	2017	2018	2019	2020
1	ADANIPORTS.NS	0.00336	0.48156	-0.02990	-0.05913	0.28095
2	TECHM.NS	-0.05990	0.03289	0.43739	0.06008	0.27700
3	BRITANNIA.NS	-0.03352	0.63150	0.31500	-0.02425	0.17657
4	TCS.NS	-0.02104	0.14392	0.43109	0.13606	0.32070
5	SUNPHARMA.NS	-0.22752	-0.09920	-0.25007	-0.00231	0.36392
6	INFY.NS	-0.08564	0.04091	0.27512	0.09939	0.70428
7	HINDALCO.NS	0.82627	0.72206	-0.16578	-0.02963	0.12249
8	DR REDDY.NS	-0.01551	-0.21703	0.08833	0.10261	0.80770
9	HCLTECH.NS	-0.02116	0.07483	0.09747	0.18373	0.65425
10	HINDUNILVR.NS	-0.03526	0.65730	0.35340	0.06768	0.23694
11	TATASTEEL.NS	0.52001	0.80061	-0.24273	-0.08473	0.37606
12	NTPC.NS	0.14014	0.07403	-0.15577	-0.03928	-0.18264
13	CIPLA.NS	-0.13207	0.07509	-0.15045	-0.08575	0.72295
14	ITC.NS	0.10680	0.09255	0.07377	-0.15918	-0.12222
15	BPCL.NS	0.41845	0.21241	-0.28677	0.33864	-0.22486
16	WIPRO.NS	-0.14736	0.33284	0.04517	0.00331	0.55935
17	JSWSTEEL.NS	0.58471	0.65086	0.15406	-0.11153	0.44424
18	GAIL.NS	0.18783	0.53097	-0.03660	-0.33149	0.01691
19	DIVISLAB.NS	-0.32585	0.37664	0.34903	0.25101	1.11227
20	ULTRACEMCO.NS	0.15092	0.28156	-0.06229	0.00919	0.30090
21	NESTLEIND.NS	0.04204	0.31744	0.40971	0.34229	0.24435
22	ASIANPAINT.NS	0.01400	0.28061	0.20059	0.30141	0.54166
23	EICHERMOT.NS	0.25515	0.34704	-0.22530	-0.02931	0.14646

24	IOC.NS	0.50053	0.18659	-0.29709	-0.08824	-0.27789
25	GRASIM.NS	0.14491	0.72335	-0.27829	-0.10780	0.24963
26	BAJAJ-AUTO.NS	0.04535	0.28289	-0.17302	0.16810	0.09331
27	HDFC.NS	0.00330	0.40531	0.16903	0.20087	0.05123
28	M&M.NS	-0.06374	0.22150	0.07950	-0.31267	0.34290
29	UPL.NS	0.46928	0.15798	0.00331	0.16357	-0.20722
30	SHREECEM.NS	0.29428	0.24501	-0.03865	0.19977	0.18176
31	MARUTI.NS	0.14683	0.78001	-0.22653	-0.01447	0.04621
32	TITAN.NS	-0.07172	1.56970	0.09215	0.28012	0.35713
33	BAJAJFINSV.NS	0.45924	0.77323	0.25235	0.44027	-0.05034
34	KOTAKBANK.NS	-0.01038	0.41326	0.25656	0.34700	0.19208
35	COALINDIA.NS	-0.10193	-0.12365	-0.09713	-0.12484	-0.36093
36	BHARTIARTL.NS	-0.10235	0.70306	-0.40803	0.55373	0.12442
37	HEROMOTOCO.NS	0.13313	0.24900	-0.17200	-0.21887	0.27849
38	LT.NS	0.04646	0.38818	0.14028	-0.09966	-0.01706
39	ONGC.NS	0.18361	0.01534	-0.22069	-0.13266	-0.26991
40	SBIN.NS	0.09833	0.27217	-0.03647	0.11399	-0.17790
41	RELIANCE.NS	0.06604	0.69763	0.23248	0.35062	0.31512
42	BAJFINANCE.NS	0.38160	1.01908	0.53284	0.59390	0.25144
43	AXISBANK.NS	0.00011	0.25952	0.09620	0.20214	-0.17130
44	HDFCBANK.NS	0.10788	0.56411	0.14408	0.18440	0.12334
45	TATAMOTORS.NS	0.17442	-0.11334	-0.59312	0.06745	-0.00325
46	ICICIBANK.NS	-0.02928	0.37091	0.16065	0.48151	-0.00317
47	POWERGRID.NS	0.30142	0.09093	-0.00774	-0.04875	-0.02890

48	INDUSINDBK.NS	0.14945	0.51630	-0.01512	-0.05365	-0.39706

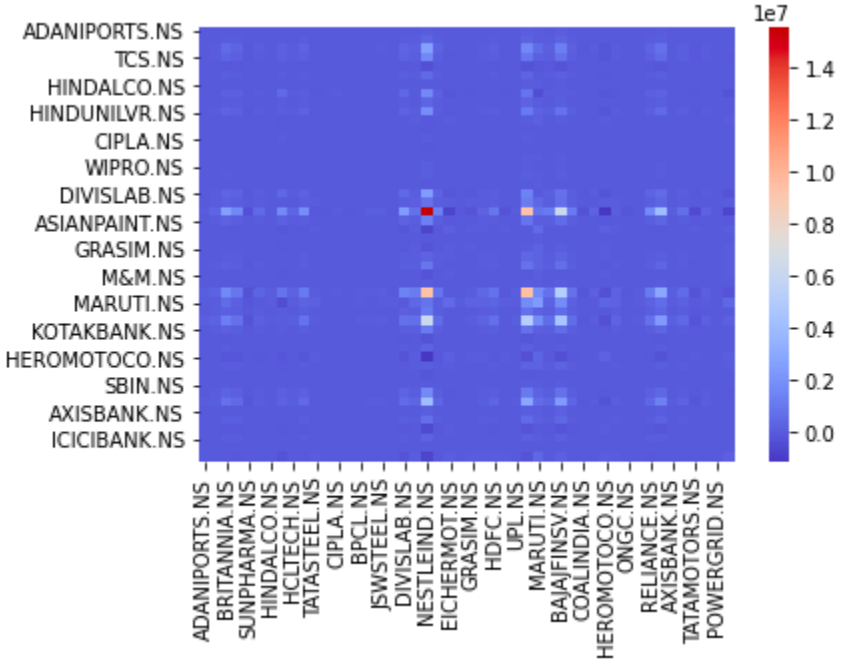
Assuming the distribution we have estimated its expected return, variance and standard deviation:

S. No	Stock	Mean	Variance	Standard Deviation
1	ADANIPORTS.NS	0.135368	4.170005e+03	64.575576
2	TECHM.NS	0.149490	2.048748e+04	143.134486
3	BRITANNIA.NS	0.213061	5.744916e+05	757.952255
4	TCS.NS	0.202147	2.243716e+05	473.678846
5	SUNPHARMA.NS	-0.043034	1.741845e+04	131.978978
6	INFY.NS	0.206812	2.784870e+04	166.879289
7	HINDALCO.NS	0.295084	2.376036e+03	48.744603
8	DRREDDY.NS	0.153219	5.453564e+05	738.482478
9	HCLTECH.NS	0.197825	1.317526e+04	114.783525
10	HINDUNILVR.NS	0.256013	2.444954e+05	494.464716
11	TATASTEEL.NS	0.273843	1.515546e+04	123.107530
12	NTPC.NS	-0.032703	3.201193e+02	17.891878
13	CIPLA.NS	0.085954	6.613395e+03	81.322786
14	ITC.NS	-0.001656	1.518338e+03	38.965857
15	BPCL.NS	0.091577	4.629948e+03	68.043722
16	WIPRO.NS	0.158663	1.876900e+03	43.323201
17	JSWSTEEL.NS	0.344468	4.997870e+03	70.695618
18	GAIL.NS	0.073524	1.037596e+03	32.211741
19	DIVISLAB.NS	0.352619	5.099136e+05	714.082322

20	ULTRACEMCO.NS	0.136057	2.180140e+05	466.919691
21	NESTLEIND.NS	0.271163	1.557141e+07	3946.063168
22	ASIANPAINT.NS	0.267651	1.326236e+05	364.175182
23	EICHERMOT.NS	0.098809	2.206931e+05	469.779865
24	IOC.NS	0.004781	1.676282e+03	40.942425
25	GRASIM.NS	0.146362	3.652424e+04	191.113151
26	BAJAJ-AUTO.NS	0.083329	6.564113e+04	256.205246
27	HDFC.NS	0.165948	1.232879e+05	351.123752
28	M&M.NS	0.053499	1.429273e+04	119.552190
29	UPL.NS	0.117383	8.030829e+03	89.614892
30	SHREECEM.NS	0.176431	9.473662e+06	3077.931465
31	MARUTI.NS	0.146411	2.398800e+06	1548.806003
32	TITAN.NS	0.445474	1.079785e+05	328.600860
33	BAJAJFINSV.NS	0.374949	4.482049e+06	2117.085119
34	KOTAKBANK.NS	0.239704	1.034109e+05	321.575621
35	COALINDIA.NS	-0.161697	3.989233e+03	63.160378
36	BHARTIARTL.NS	0.174166	6.979714e+03	83.544682
37	HEROMOTOCO.NS	0.053950	2.561427e+05	506.105461
38	LT.NS	0.091641	4.478029e+04	211.613549
39	ONGC.NS	-0.084862	1.445741e+03	38.022904
40	SBIN.NS	0.054022	2.435393e+03	49.349706
41	RELIANCE.NS	0.332378	2.264325e+05	475.849238
42	BAJFINANCE.NS	0.555770	1.374807e+06	1172.521635
43	AXISBANK.NS	0.077334	1.226295e+04	110.738182

44	HDFCBANK.NS	0.224761	5.646520e+04	237.624076
45	TATAMOTORS.NS	-0.093567	2.099531e+04	144.897601
46	ICICIBANK.NS	0.196126	8.041118e+03	89.672281
47	POWERGRID.NS	0.061392	4.065173e+02	20.162275
48	INDUSINDBK.NS	0.039986	1.778378e+05	421.708200

The link for variance matrix is : https://bit.ly/3fReQY8
The variance covariance heatmap is given below:

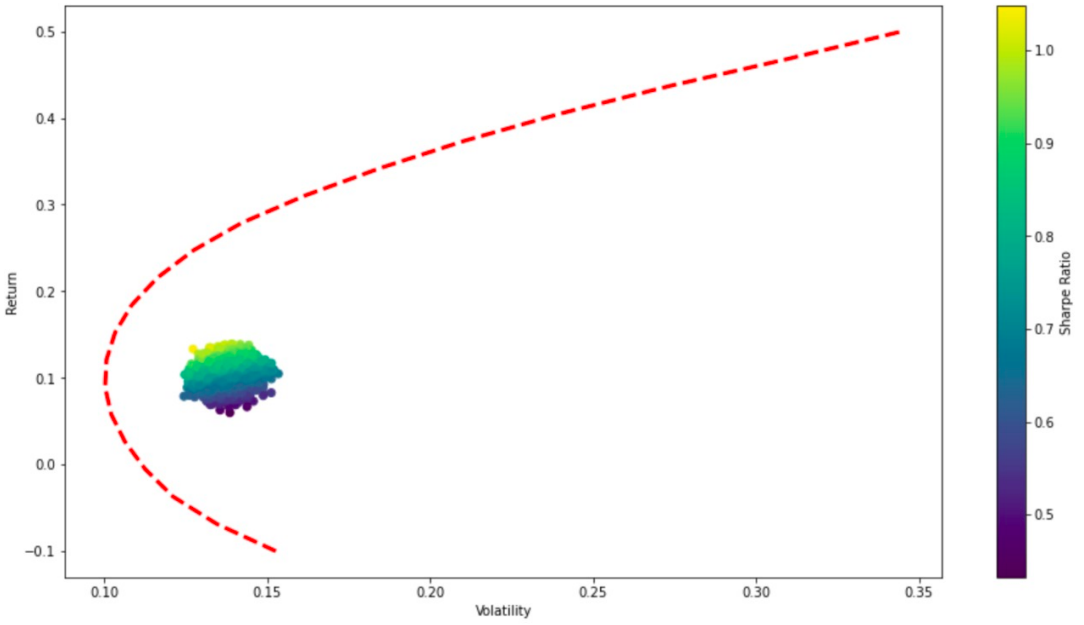


TASK 4:
(CODE: https://bit.ly/3wz6KcB)
MARKOWITZ PORTFOLIO CONSTRUCTION:

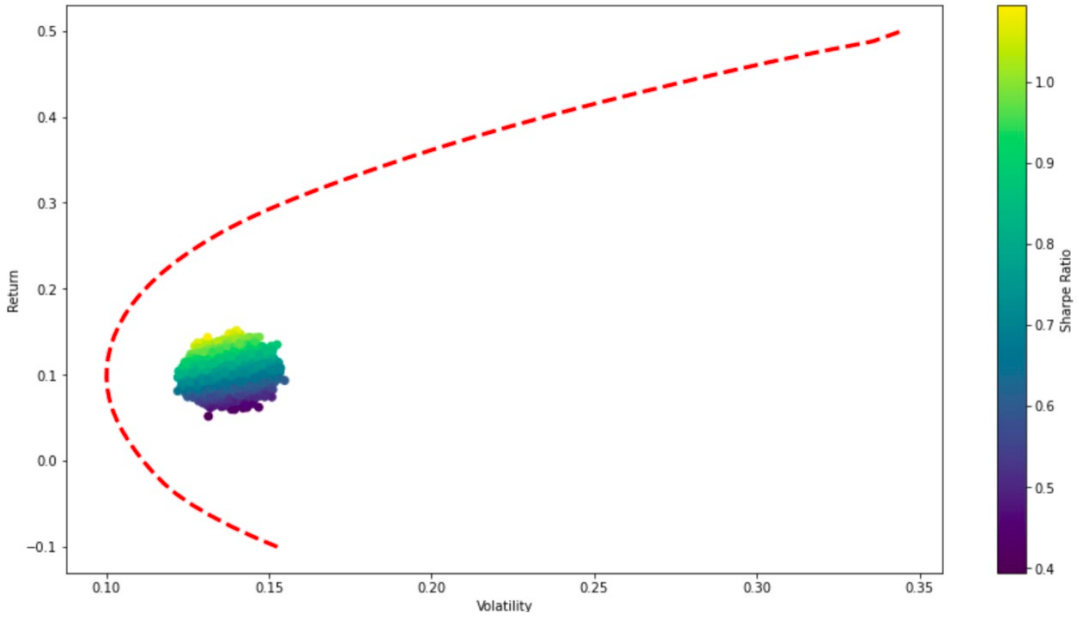
The aim of the Markowitz portfolio construction is maximising return for a given risk. This method involves an approach aiming for a portfolio with maximum return with minimum associated risk. The idea used for code is initialising random weights and then maximising the sharpe ratio. With this finally we reach at optimal weights.

To find these optimal weights we had made sample portfolios. It is observed that for a number around 48, we need an extremely large number of sample portfolios. (We had tried a maximum of 360000 sample size which took us a day to compute the scatter shown below, yet it is not sufficient as it can be seen in the efficient curve plot along with the samples.) This mainly occurs because returns are calculated at a shorter horizon and too many stocks need more computational power.

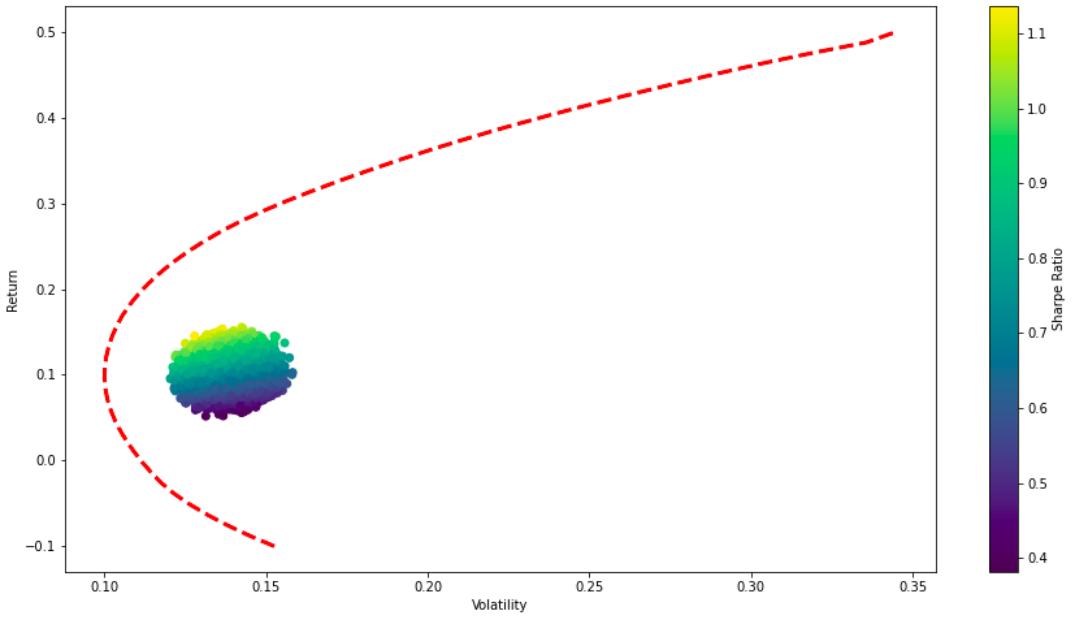
First Try: 6 Thousand sample portfolios



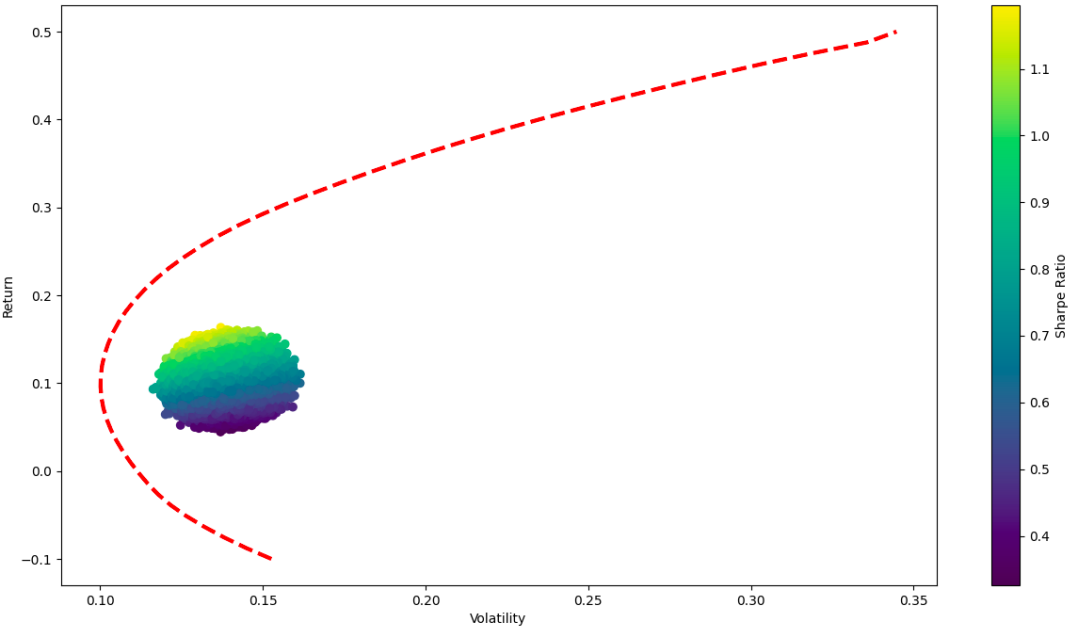
Second Try: 60 Thousand sample portfolios



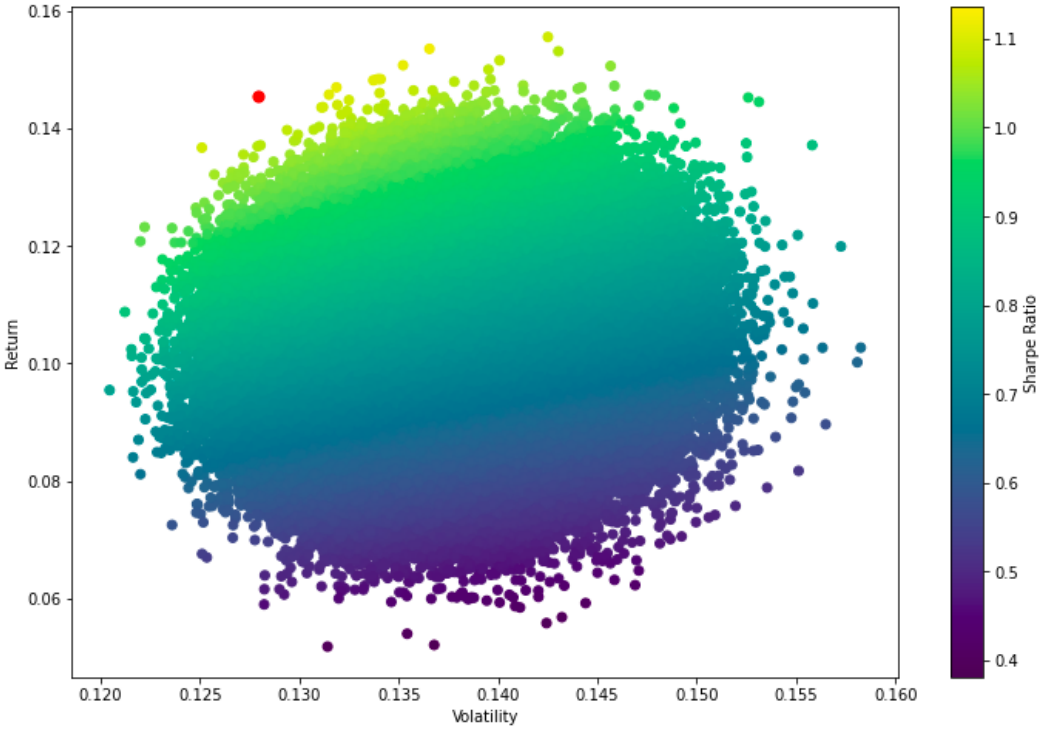
Third Try: 0.6 million sample portfolios



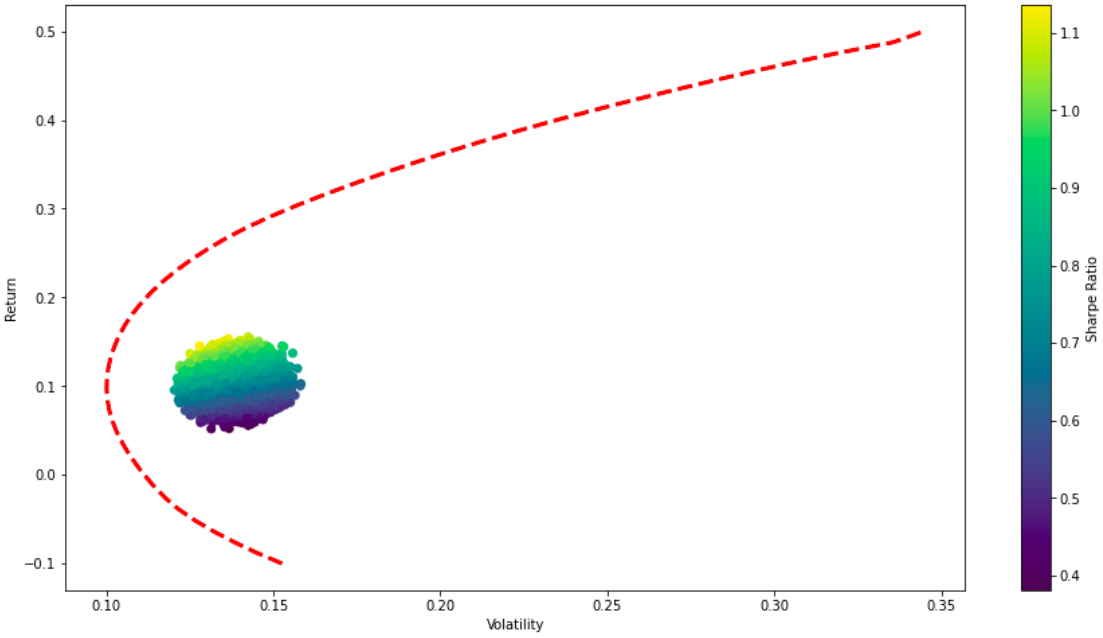
Fourth Try: 36 million sample portfolios



Scatter Plot of 36 million sample portfolios

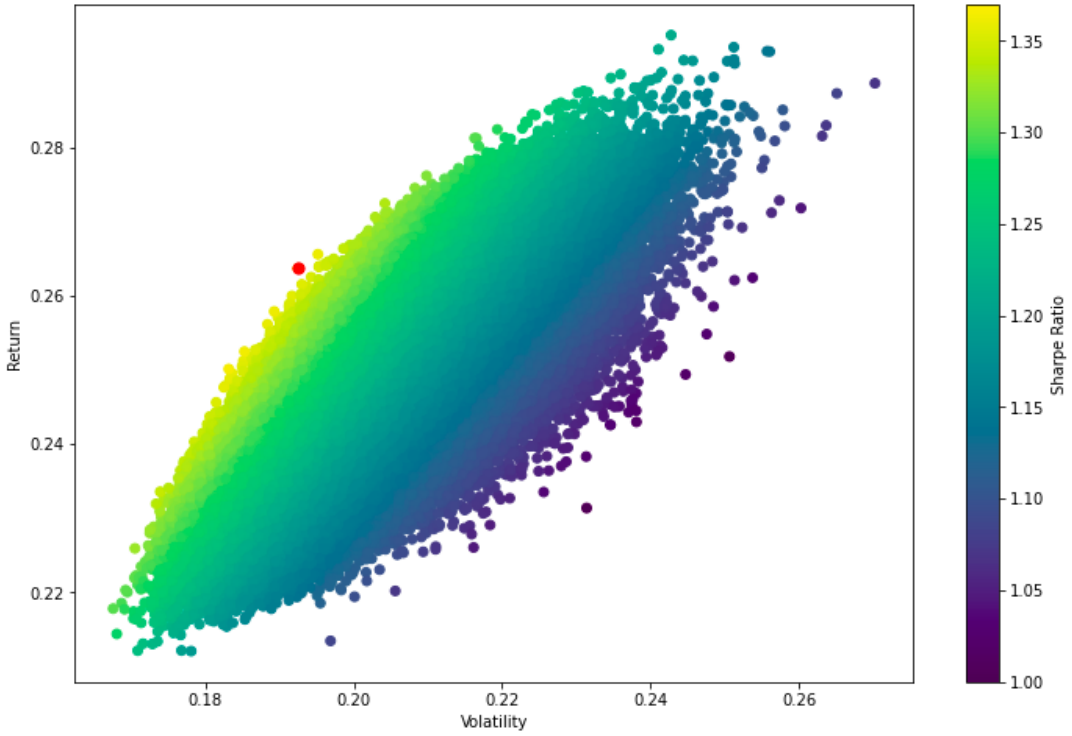


Efficient Frontier for Scatter Plot of 48 stock portfolio with 36 million samples

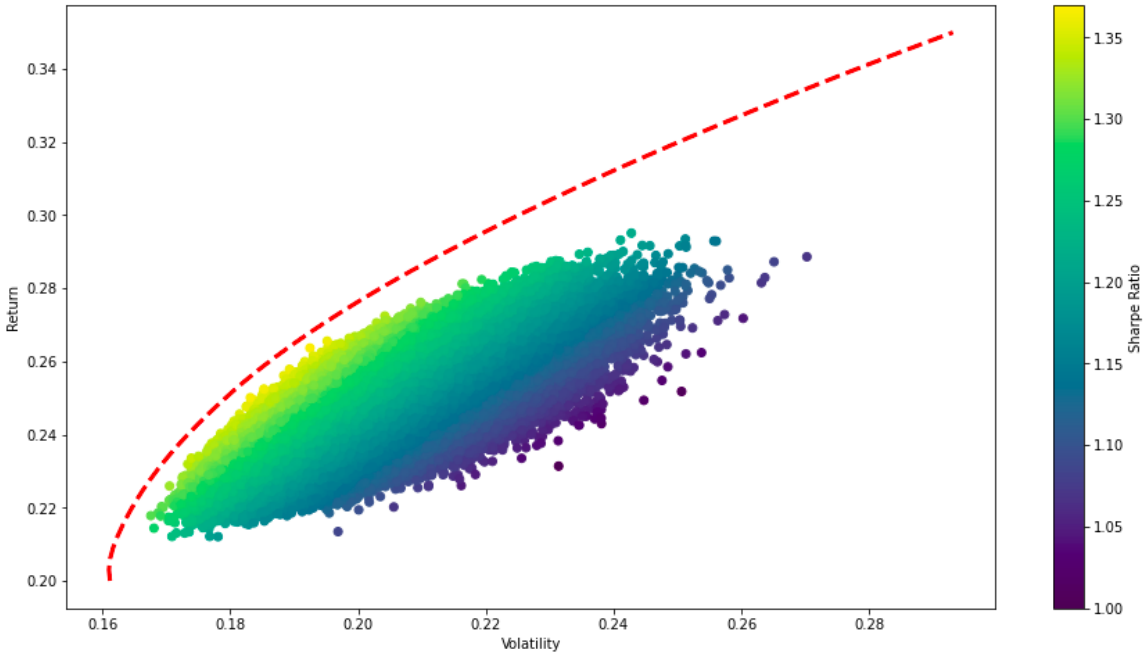


When we ran the same for a smaller size of 15 stocks with the highest sharpe ratio, for annual returns, the observations made were much better. The code for obtaining these fifteen stocks can be found: https://bit.ly/3uw6ERh

Scatter Plot of 15 stocks portfolio for 0.6 million sample portfolios



Efficient Frontier for Scatter Plot of 15 stock portfolio with 0.6 million samples



Weights for the optimal portfolio according to Markowitz method

S. No.	Stock	Weights
1	ADANIPORTS.NS	1.03126597e-02
2	TECHM.NS	1.94616147e-02
3	BRITANNIA.NS	3.49119588e-02
4	TCS.NS	1.56493259e-02
5	SUNPHARMA.NS	1.30501423e-02
6	INFY.NS	3.74950719e-02
7	HINDALCO.NS	1.07189485e-03
8	DRREDDY.NS	2.43446019e-02
9	HCLTECH.NS	2.43809269e-02
10	HINDUNILVR.NS	4.63044568e-02
11	TATASTEEL.NS	7.06716562e-03
12	NTPC.NS	2.08537294e-02
13	CIPLA.NS	3.05785225e-02
14	ITC.NS	4.30175939e-03
15	BPCL.NS	3.97881879e-03
16	WIPRO.NS	1.64804829e-02
17	JSWSTEEL.NS	1.73843193e-02
18	GAIL.NS	2.52135949e-02
19	DIVISLAB.NS	1.74008150e-02
20	ULTRACEMCO.NS	7.44642454e-03
21	NESTLEIND.NS	3.97722985e-02

22	ASIANPAINT.NS	4.37907180e-02
23	EICHERMOT.NS	2.40419257e-02
24	IOC.NS	1.82971564e-03
25	GRASIM.NS	3.09149387e-02
26	BAJAJ-AUTO.NS	4.05224275e-02,
27	HDFC.NS	4.06320418e-02
28	M&M.NS	1.48797450e-04
29	UPL.NS	2.82335003e-03
30	SHREECEM.NS	3.34225753e-02
31	MARUTI.NS	2.43515307e-03
32	TITAN.NS	4.39188067e-02
33	BAJAJFINSV.NS	4.41411218e-02
34	KOTAKBANK.NS	5.03872465e-03
35	COALINDIA.NS	4.30005262e-03
36	BHARTIARTL.NS	3.10165906e-02
37	HEROMOTOCO.NS	3.27807032e-02
38	LT.NS	1.00451472e-02
39	ONGC.NS	2.84610078e-02
40	SBIN.NS	4.95647661e-04
41	RELIANCE.NS	4.47100406e-02
42	BAJFINANCE.NS	4.44272119e-02
43	AXISBANK.NS	6.11015787e-03
44	HDFCBANK.NS	1.16723194e-02
45	TATAMOTORS.NS	6.96014902e-04

46	ICICIBANK.NS	3.43925413e-02
47	POWERGRID.NS	1.96844898e-02
48	INDUSINDBK.NS	8.71943363e-05

PROOF OF WHY EFFICIENT FRONTIER IS PARABOLA :

Suppose there are **n** risky securities with the expected return on the i^{th} security denoted by $\boldsymbol{\alpha}_i$
covariance of returns between the i^{th} and j^{th} security denoted by $\boldsymbol{\sigma}_{ij}$;
the variance of the return on the i^{th} security is denoted by $\boldsymbol{\sigma}_i^2$.
As all securities are assumed to be risky thus the covariance matrix is non-singular.
Amongst all the feasible portfolios the locus with minimum variance for prescribed return is our frontier.

Let $\mathbf{w_i}$ be the weight of individual security in the portfolio.
Also, $\sum \mathbf{w_i} = \mathbf{1}$
Then, the frontier can be described as the set of portfolios which satisfy the constrained minimization problem,
System 1 is

$$\text{Min } (\tfrac{1}{2} \sum \sum \mathbf{w_i} * \mathbf{w_j} * \boldsymbol{\sigma}_{ij})$$

$$\boldsymbol{\alpha} = \sum \mathbf{w_i} * \boldsymbol{\alpha}_i$$

$$\sum \mathbf{w_i} = \mathbf{1}$$

Using the Lagrangian multiplier the minimization can be written as:
System 2 is

$$\mathbf{min}(\tfrac{1}{2} \sum \sum \mathbf{w_i} * \mathbf{w_j} * \boldsymbol{\sigma}_{ij} + \lambda * (\boldsymbol{\alpha} - \sum \mathbf{w_i} * \boldsymbol{\alpha}_i) + \mu * (1 - \sum \mathbf{w_i}))$$

where $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$ are the multipliers. A critical point occurs where the partial derivatives of (2) with respect $\mathbf{w_1}\ \mathbf{w_2}$ ..... $\mathbf{w_n}$, $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$ are equal to zero.
So, System 3 is

- $\sum \mathbf{w_i} * \boldsymbol{\sigma}_{ij} - \boldsymbol{\lambda} * \mathbf{w_i} - \boldsymbol{\mu} = 0$ for i = 1, 2, 3 . . n
- $\boldsymbol{\alpha} - \sum \mathbf{w_i} * \boldsymbol{\alpha}_i = \mathbf{0}$

- $\sum \mathbf{w_i} - 1 = 0$

Further, the weights which satisfy system 3 minimize system 1 and are unique by the assumption on the covariance matrix. System 3 is linear in the $\mathbf{w_i}$'s and hence, we have from (3a) that
System 4 is

$$\mathbf{w_k} = \lambda \sum \mathbf{v_{kj}} * \boldsymbol{a}_j + \mu \sum \mathbf{v_{kj}} \quad k=1...n$$

where the v_{ij} are defined as the elements of the inverse of the variance-covariance matrix, i.e. Covariance inverse, $\boldsymbol{\Omega}^{-1} = [\mathbf{v_{ij}}]$. Multiplying system 4 by $\boldsymbol{a_k}$ and summing over k=1, . . ., n, we have that

$$\sum \boldsymbol{a_k} \mathbf{w_k} = \lambda \sum \sum \boldsymbol{a_k} \mathbf{v_{kj}} * \boldsymbol{a_j} + \mu \sum \sum \boldsymbol{a_k} \mathbf{v_{kj}} \quad \mathbf{k=1...n \ldots (5)}$$

And by summing over 1 we have

$$\sum \mathbf{w_k} = \lambda \sum \sum \mathbf{v_{kj}} * \boldsymbol{a_j} + \mu \sum \sum \mathbf{v_{kj}} \quad \mathbf{k=1...n \ldots (6)}$$

Let

- A = $\sum \sum \mathbf{v_{kj}} * \boldsymbol{a}_j$
- B = $\sum \sum \mathbf{v_{kj}} * \boldsymbol{a}_j * \boldsymbol{a}_k$
- C = $\sum \sum \mathbf{v_{kj}}$

Here B and C >0

System 3 now will be:
System 7
$\alpha = B*\lambda + A*\mu$
$1 = A*\lambda + C*\mu$

Solving the above two equations:

System 8:
$\lambda = (C*\alpha - A)/D$
$\mu = (B - A*\alpha)/D$

$D = BC - A^2$

Because $\boldsymbol{\Omega}^{-1}$ is positive definite,
$\sum \sum v_{ij}(B\boldsymbol{a}_i - A)(B\boldsymbol{a}_j - A) = B^2C - 2A^2B + A^2B = B(BC - A^2) = BD.$
But B > 0, hence D > 0.

We can now substitute for **ƛ** and **μ** system 8 into system 4 to solve for the proportions of each risky asset held in the frontier portfolio with the expected return that is:

$$w_k = (\sum v_{kj}(Ca_j - A) + \sum v_{kj}(B - Aa_j))/D \ldots (9)$$

$$\sum\sum w_i * w_j * \sigma_{ij} = ƛ*(\sum w_i * a_i) + \mu*(\sum w_i) \ldots (10)$$

$$\text{Let } \sum\sum w_i * w_j * \sigma_{ij} = \sigma^2$$

$$\text{Thus } \sigma^2 = ƛa + \mu \ldots (11)$$

Substituting for **ƛ** and μ from (8) into (11) we write the equation for the variance of a frontier portfolio as a function of its expected return, as

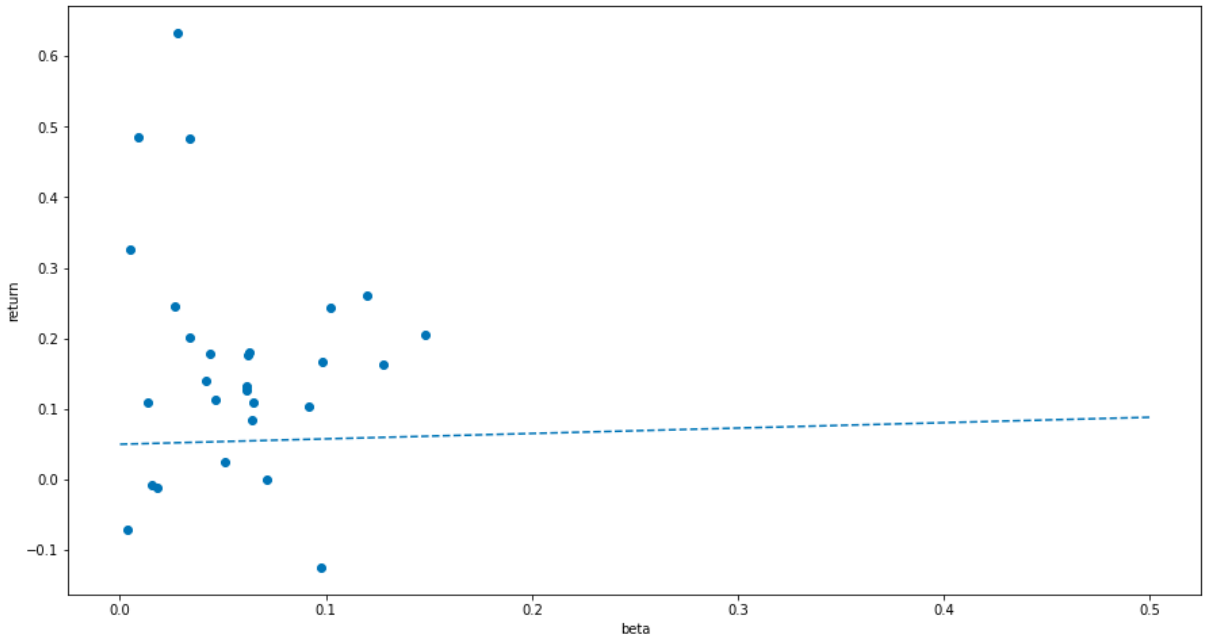
$$\sigma^2 = (C\alpha^2 - 2A\alpha + B)/D$$

Thus, the frontier in mean-variance space is a parabola.

TASK 5:
(CODE: https://bit.ly/3t5il1a)

Using a risk-free rate of 5% and we have plotted the Security Market Line (SML).

SML is a graphical representation of CAPM and helps us predict underpriced and overpriced stocks. If a stock is plotted on the SML plot and it appears above the SML, it is considered undervalued as it indicates that the stock offers a greater return against risk.
Conversely, if the stock is below the SML, it is considered overvalued in price since the return does not account for the risk. For our Nifty 50 stocks, some of the observations of underpriced and overpriced stocks is shown below:



Overpriced

DRREDDY, NTPC, CIPLA, ITC, HEROMOTOCO, TATAMOTORS

Underpriced

ADANIPORTS, TECHM, TCS, HINDALCO, GAIL, ULTRACEMCO

Observation: BAJFINANCE as the highest return with a pretty less beta.

TASK 6 :
(CODE: https://bit.ly/3uzIXay)

To read about another approach to estimate variance-covariance matrix and implementing the same.

Maximum likelihood estimator(MLE) or also known as empirical covariance is used to estimate the covariance matrix provided that the number of observations is high enough in comparison to the number of features (the variables describing the

observations). In our case, we are estimating the covariance matrix of returns of different stocks. For this we required a probability model for optimizing this large data we have in our matrix. Estimated data is more consistent than the observed data which we have in our space, making us employ the MLE to our observed covariance matrix.

MAXIMUM LIKELIHOOD ESTIMATION
Let us assume that our distribution has random samples such as X = (X1, X2, . . . , Xn). We want to estimate after obtaining accurate data from F.
The function **f (·|θ)** determines a density for a given value of θ.
The method of maximum likelihood estimation is a useful method of estimating θ.
Now, we want to define a function named as likelihood that tells us the probability of any particular θ given to us from the actual data and that function will be called as **L(θ|X).** And this likelihood is defined as -

$$\mathbf{L(\theta|X)} = \prod_{i=1}^{n} f\,(Xi\,|\theta)$$

Now we will use the above described method to estimate our covariance matrix and do the implementation part.
The empirical covariance matrix can be calculated using the package. We are using the scikit-learn library of the Python programming language to deduce the empirical covariance function and further estimating by fitting our matrix into this empirical covariance function.

The link to real covariance matrix: https://bit.ly/3rUe2nH
The link to MLE estimated matrix: https://bit.ly/39UpWI2

Here is the snapshot of a real and estimated covariance matrix using MLE.

```
real_cov
```

```
array([[4.58991123e-04, 5.58997405e-05, 7.94148937e-05, ...,
        1.48820567e-04, 4.99662365e-05, 1.12802653e-04],
       [5.58997405e-05, 2.95439696e-04, 4.35049074e-05, ...,
        4.18873342e-05, 1.16696376e-05, 3.58545801e-05],
       [7.94148937e-05, 4.35049074e-05, 2.18138685e-04, ...,
        6.33394986e-05, 1.98534914e-05, 8.22886368e-05],
       ...,
       [1.48820567e-04, 4.18873342e-05, 6.33394986e-05, ...,
        3.82322546e-04, 5.77949662e-05, 9.81136277e-05],
       [4.99662365e-05, 1.16696376e-05, 1.98534914e-05, ...,
        5.77949662e-05, 1.76804846e-04, 3.73201112e-05],
       [1.12802653e-04, 3.58545801e-05, 8.22886368e-05, ...,
        9.81136277e-05, 3.73201112e-05, 3.21014781e-04]])
```

```
cov.covariance_
```

```
array([[4.23099237e-04, 4.39252580e-05, 6.46404374e-05, ...,
        1.57578183e-04, 4.80649039e-05, 9.97012832e-05],
       [4.39252580e-05, 2.80682048e-04, 2.09861871e-05, ...,
        5.76606174e-05, 8.47544835e-07, 4.27417321e-06],
       [6.46404374e-05, 2.09861871e-05, 2.02851310e-04, ...,
        7.80779802e-05, 1.44054638e-05, 8.50900202e-05],
       ...,
       [1.57578183e-04, 5.76606174e-05, 7.80779802e-05, ...,
        4.52470620e-04, 7.08224027e-05, 1.20055280e-04],
       [4.80649039e-05, 8.47544835e-07, 1.44054638e-05, ...,
        7.08224027e-05, 2.01443203e-04, 4.62932186e-05],
       [9.97012832e-05, 4.27417321e-06, 8.50900202e-05, ...,
        1.20055280e-04, 4.62932186e-05, 2.82798377e-04]])
```

TASK 7:
WHAT IF THE RETURNS DO NOT FOLLOW NORMAL DISTRIBUTION

- SUITABLE RETURN:

When calculated from just the previous day's closing price, returns do not yield a normal curve, but if we increase the span to a week as we have tried doing in our project, the distribution is somewhat nearly normal. On expanding the span further, the distribution is likely to form a better normal curve.

- SUITABLE/ALTERNATIVE RISK MEASURE:

The estimation of risk plays a crucial role in portfolio management. In the cases where we assume the returns to follow a normal distribution, variance is termed as the risk and calculated straight forward for a given N (μ, σ), as σ^2
In our study of finding the returns distribution, we observed the normal distribution was significant but was not always the case. It could be estimated to nearly normal but not precisely. In such cases, better estimation of risk, i.e., a better variance of the distribution, is required. According to various studies, it is reported that at the confidence level of 95%, this simple historic variance is not precise. Indeed, different approaches like exponentially weighted moving average must be considered to estimate the variance better.
If a distribution is not normal, we can find the skewness and kurtosis, along with variance, volatility calculated using rolling window moving average (MA) and exponential weighted moving average (EMWA).

Table 1. Stochastic parameters for MSE

ISIN Code	Kurt	σ	σ_σ MA	σ_σ EWMA
ALK	4.10	0.0232	0.00856	0.00791
BESK	7.40	0.0270	0.01265	0.00903
GRNT	3.31	0.0271	0.01070	0.00825
KMB	4.54	0.0227	0.00875	0.00770
MPT	7.21	0.0256	0.00777	0.00806
REPL	27.27	0.0213	0.00937	0.00785
SBT	9.79	0.0210	0.00615	0.00684
STIL	32.55	0.0342	0.01907	0.01361
MTUR	8.72	0.0190	0.00551	0.00521
TPFL	5.41	0.0242	0.00874	0.00806
MBI	5.61	0.0167	0.00548	0.00636
MSE	10.54	0.0238	0.00934	0.00808

Note: Column headings are as follows: (1) Stock code; (2): Kurtosis of the daily return series; (3): Volatility of the daily return series; (4): Volatility of the volatility sequence calculated using rolling window moving average; and (5): Volatility of the volatility sequence calculated using exponentially weighted moving average.

Snippet from a research paper

In the above snapshot, variance is significantly different from volatility when computed using MA and EWMA, which are pretty close to each other.
Thus we can alternatively use MA and EWMA for precise risk evaluation.

Another concern associated with this approach is that it may work for the confidence level of 95% but cannot be relied on for a level of 99%. This dodges a question on the reliability of variance as a parameter for risk estimation in a portfolio.

Research is undertaken to look into new avenues to estimate variance further because risk management is an essential component of portfolio management.

Once better returns and risk are estimated, weight identification can be made on a similar Markowitz approach. This time the weights should be better assigned as more specific risk and returns are assessed.

References:

Task 1:
https://community.esri.com/t5/python-documents/creating-multiple-graphs-per-page-using-matplotlib/ta-p/916434
https://machinelearningmastery.com/decompose-time-series-data-trend-seasonality/
https://towardsdatascience.com/trend-seasonality-moving-average-auto-regressive-model-my-journey-to-time-series-data-with-edc4c0c8284b
https://algorithmia.com/blog/introduction-to-time-series
Task 2:
https://math.hws.edu/javamath/ryan/ChiSquare.html#:~:text=Calculate%20the%20chi%20square%20statistic,E)2%20%2F%20E%20%5D.
Check for normal distribution, Shapiro Wilk test
https://towardsdatascience.com/6-ways-to-test-for-a-normal-distribution-which-one-to-use-9dcf47d8fa93
Identifying Data's Distribution, chi-square statistics
https://towardsdatascience.com/identify-your-datas-distribution-d76062fc0802
Task 4:
https://towardsdatascience.com/python-markowitz-optimization-b5e1623060f5
Proof of why the curve will be a parabola.
http://dspace.mit.edu/bitstream/handle/1721.1/46832/analyticderivati00mert.pdf?sequence=1
Task 5:
https://www.wallstreetmojo.com/security-market-line/
https://www.quantconnect.com/tutorials/introduction-to-financial-python/market-risk
Task 6:
https://www.coursera.org/learn/advanced-portfolio-construction-python
https://en.wikipedia.org/wiki/Maximum_likelihood_estimation
https://scikit-learn.org/stable/modules/generated/sklearn.covariance.EmpiricalCovariance.html
https://scikit-learn.org/stable/modules/covariance.html#covariance
Task 7:
https://www.utmsjoe.mk/files/Vol.%206%20No.%202/UTMSJOE-2015-0602-003-Ivanovski-Stojanovski-Narasanov.pdf

https://www.diva-portal.org/smash/get/diva2:428847/FULLTEXT01.pdf