Optimal portfolio (최적 포트폴리오)

- Construct a diversified portfolio consisting of three stocks, and one market index. We are using Disney, IBM, Southwest Airlines, and S&P 500. (2007 – 2014)

time	DIS	IBM	LUV	S&P500
31JAN2007	0.0263	0.0206	-0.0144	0.0153
28FEB2007	-0.0262	-0.0596	0.0016	-0.0193
30MAR2007	0.0053	0.0142	-0.0278	0.0109
30APR2007	0.016	0.0843	-0.0238	0.044
31MAY2007	0.0132	0.0469	-0.0028	0.0345
29JUN2007	-0.0231	-0.0127	0.0422	-0.0166
31JUL2007	-0.0334	0.0513	0.0503	-0.0312
31AUG2007	0.0182	0.0582	-0.0348	0.0151
28SEP2007	0.0235	0.0095	-0.0205	0.0375
31OCT2007	0.007	-0.0143	-0.0399	0.0174
30NOV2007	-0.0427	-0.0908	-0.0042	-0.0412
31DEC2007	-0.0157	0.0278	-0.1375	-0.0061

Correlation and covariance

	DIS	IBM	LUV	S&P500
average return	0.0140	0.0082	0.0151	0.0069
varicance	0.0044	0.0029	0.0083	0.0021
std. dev.	0.0661	0.0536	0.0909	0.0461
annual vol.	0.2290	0.1856	0.3149	0.1598
correlation matrix	DIS	IBM	LUV	S&P500
DIS	1			
IBM	0.4491	1		
LUV	0.5395	0.2964	1	
S&P500	0.8297	0.5665	0.5678	1
covariance matrix	DIS	IBM	LUV	S&P500
DIS	0.0044	0.0016	0.0032	0.0025
IBM	0.0016	0.0029	0.0014	0.0014
LUV	0.0032	0.0014	0.0083	0.0024
S&P500	0.0025	0.0014	0.0024	0.0021

포트폴리오의 기대수익율과 분산

Calculate the returns and variances of a portfolio.

$$r_{portfolio} = w_{IBM}r_{IBM} + w_{LUV}r_{LUV}$$

$$Variance_{portfolio} = w^2_{IBM}Variance_{IBM} + w^2_{LUV}Variance_{LUV} +$$
$$(2\ w_{LUV}\ w_{IBM}\ Std_{LUV}\ Std_{IBM}\ Correlation_{IBM;LUV})$$

% in IBM	% in LUV	return	variance
0%	100%	1.5051%	0.00826572
10%	90%	1.4370%	0.00698376
20%	80%	1.3689%	0.00586675
30%	70%	1.3009%	0.00491469
40%	60%	1.2328%	0.00412759
50%	50%	1.1647%	0.00350545
60%	40%	1.0966%	0.00304827
70%	30%	1.0285%	0.00275604
80%	20%	0.9604%	0.00262876
90%	10%	0.8924%	0.00266644
100%	0%	0.8243%	0.00286908

Correlation이 +1 또는 -1인 경우

% in IBM	% in LUV	return	case A	case B
0%	100%	1.5051%	0.00826572	0.00826572
10%	90%	1.4370%	0.00760049	0.00584736
20%	80%	1.3689%	0.00696316	0.00384649
30%	70%	1.3009%	0.00635374	0.00226310
40%	60%	1.2328%	0.00577222	0.00109720
50%	50%	1.1647%	0.00521861	0.00034880
60%	40%	1.0966%	0.00469289	0.00001788
70%	30%	1.0285%	0.00419509	0.00010445
80%	20%	0.9604%	0.00372518	0.00060850
90%	10%	0.8924%	0.00328318	0.00153005
100%	0%	0.8243%	0.00286908	0.00286908
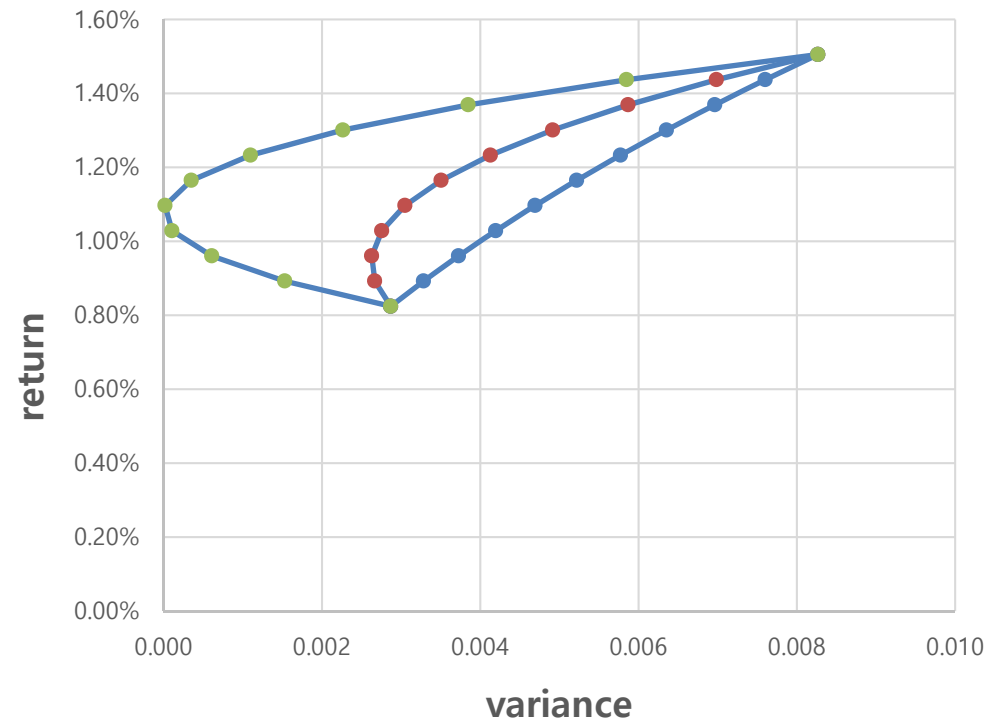
포트폴리오에 다른 주식 추가하기

IBM과 LUV 포트폴리오에 DIS 추가하기
IBM, LUV, 그리고 DIS 포트폴리오에 S&P 500 추가하기
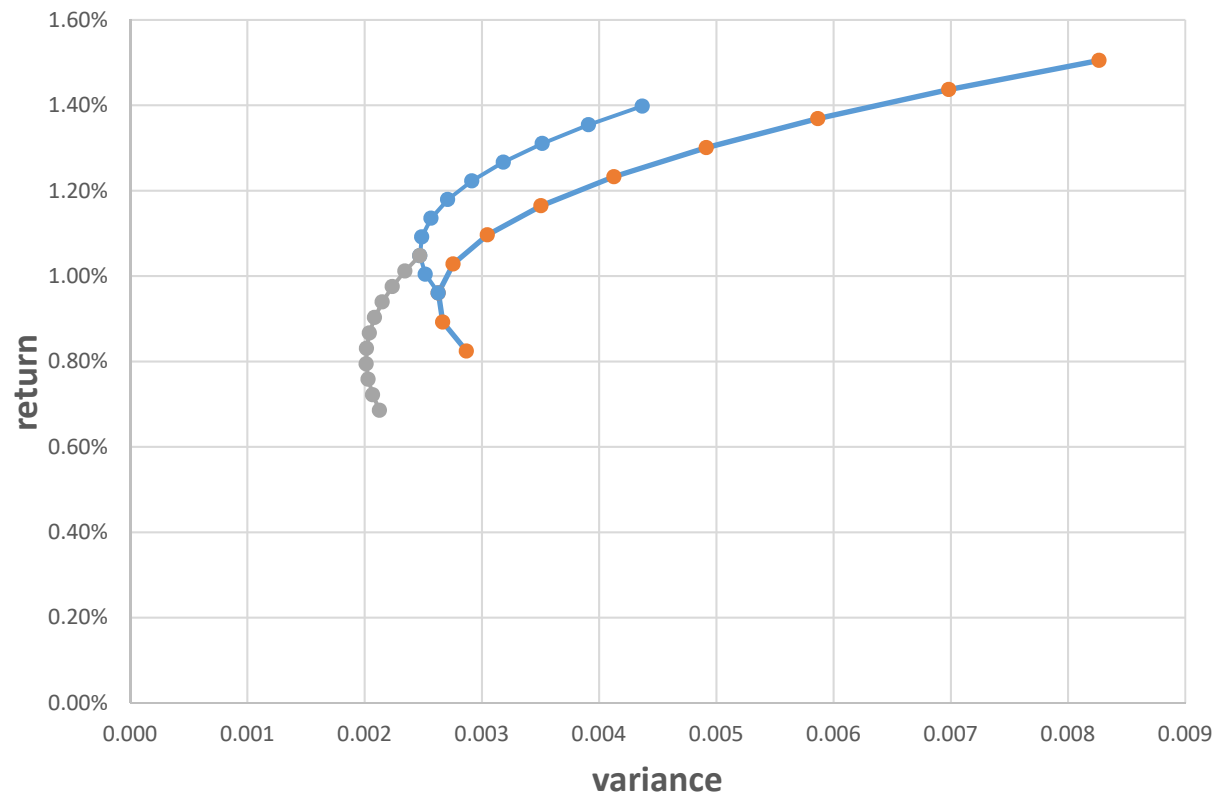
MVP, Optimal portfolio

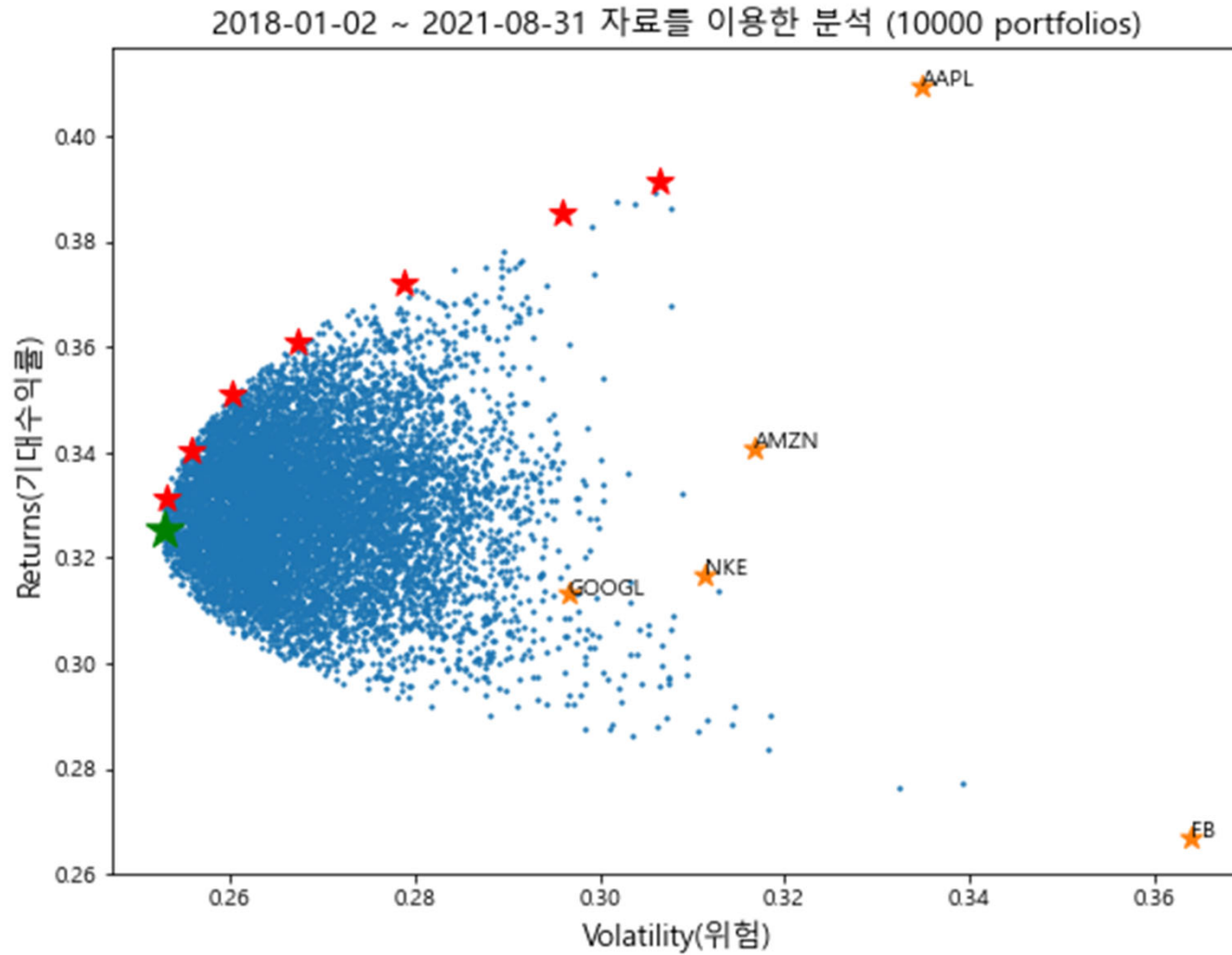EXCEL의 해찾기 기능을 이용
MMULT(), TRANSPOSE() 함수를 사용

(Q8) minimum variance portfolio, optimal portfolio				
	weight	Expected return		
DIS	25%	0.0140		
IBM	43%	0.0082		
LUV	10%	0.0151		
S&P500	23%	0.0069		
	100%			
	100%			
portfolio return		1.00%	1%	
portfolio variance	0.002254			

$$\min_{w} \ w'Vw$$

$$\text{s.t.} \quad \sum w_i = 100\%$$

$$w'e \geq 1\%$$

최적 포트폴리오 (Optimal Portfolio)

2018-01-02 ~ 2021-08-31 자료를 이용한 분석 (10000 portfolios)

(1) Assets
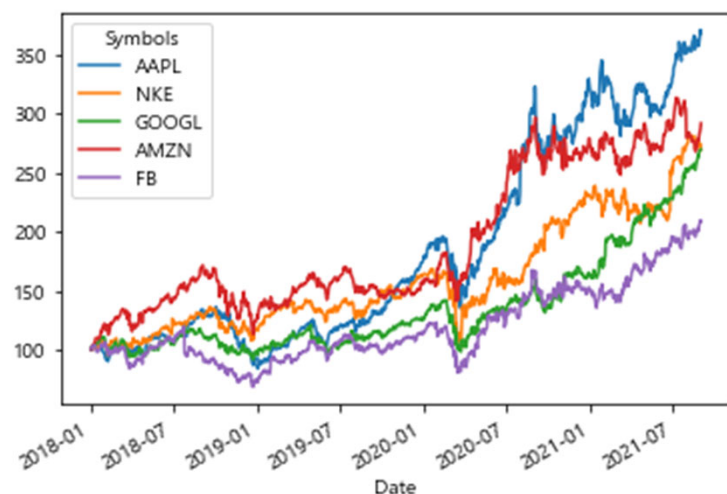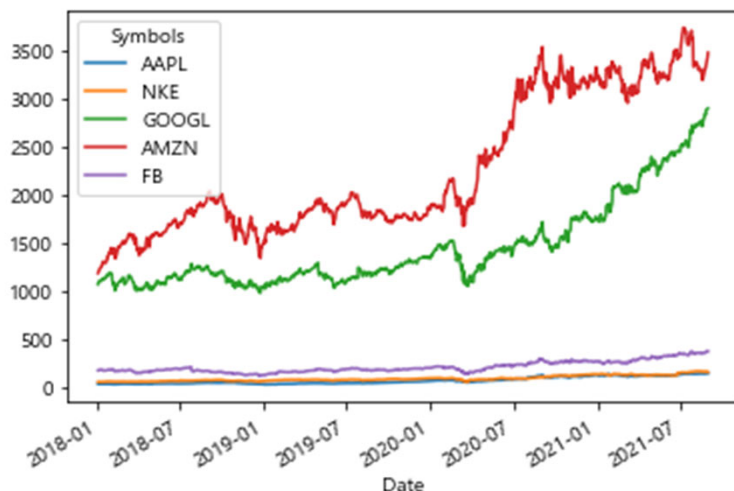
Symbols	AAPL	NKE	GOOGL	AMZN	FB
Date					
2018-01-02	41.135757	60.938103	1073.209961	1189.010010	181.419998
2018-01-03	41.128593	60.928501	1091.520020	1204.199951	184.669998
2018-01-04	41.319630	60.890110	1095.760010	1209.589966	184.330002
2018-01-05	41.790073	61.408405	1110.290039	1229.140015	186.850006
2018-01-08	41.634853	61.955486	1114.209961	1246.869995	188.279999
...	...	...	...	...	...
2022-03-25	174.720001	133.699997	2833.459961	3295.469971	221.820007
2022-03-28	175.600006	134.809998	2829.110107	3379.810059	223.589996
2022-03-29	178.960007	139.139999	2850.110107	3386.300049	229.860001
2022-03-30	177.770004	138.539993	2838.770020	3326.020020	227.850006
2022-03-31	174.610001	134.559998	2781.350098	3259.949951	222.360001

1070 rows × 5 columns

Symbols	Returns	Volatility
AAPL	0.391472	0.326006
NKE	0.232455	0.308342
GOOGL	0.266381	0.295194
AMZN	0.288000	0.323104
FB	0.124217	0.385317

(2) Portfolios

	AAPL weight	NKE weight	GOOGL weight	AMZN weight	FB weight	Returns	Volatility
0	0.202621	0.265625	0.374282	0.126116	0.031357	0.280984	0.255547
1	0.187715	0.239189	0.120621	0.176069	0.276406	0.246259	0.267633
2	0.498480	0.240327	0.129285	0.030164	0.101744	0.306771	0.270806
3	0.124861	0.282466	0.299558	0.014448	0.278667	0.233113	0.267508
4	0.011162	0.172033	0.185533	0.301706	0.329566	0.221611	0.276528
...	...	...	...	...	...	...	...
9995	0.208836	0.131949	0.281045	0.115384	0.262787	0.253163	0.272069
9996	0.082522	0.119787	0.059450	0.346604	0.391636	0.224456	0.287689
9997	0.139516	0.219743	0.215895	0.298759	0.126086	0.264912	0.259578
9998	0.016463	0.371589	0.088455	0.310880	0.212614	0.232329	0.261532
9999	0.051609	0.011416	0.281975	0.286748	0.368252	0.226296	0.292049

10000 rows × 7 columns
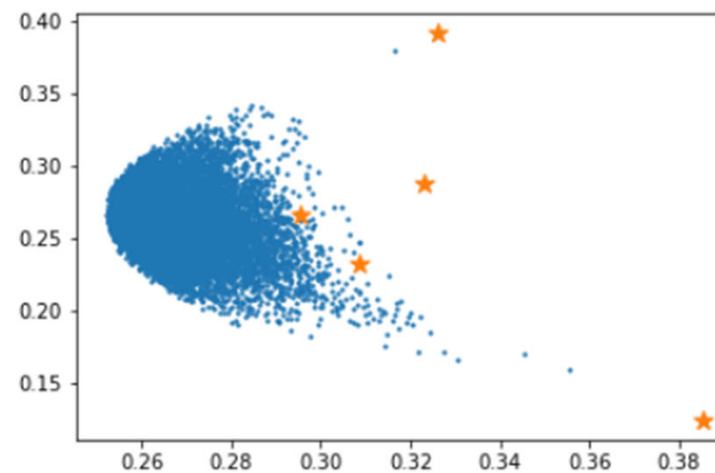
(3) Optimal portfolios
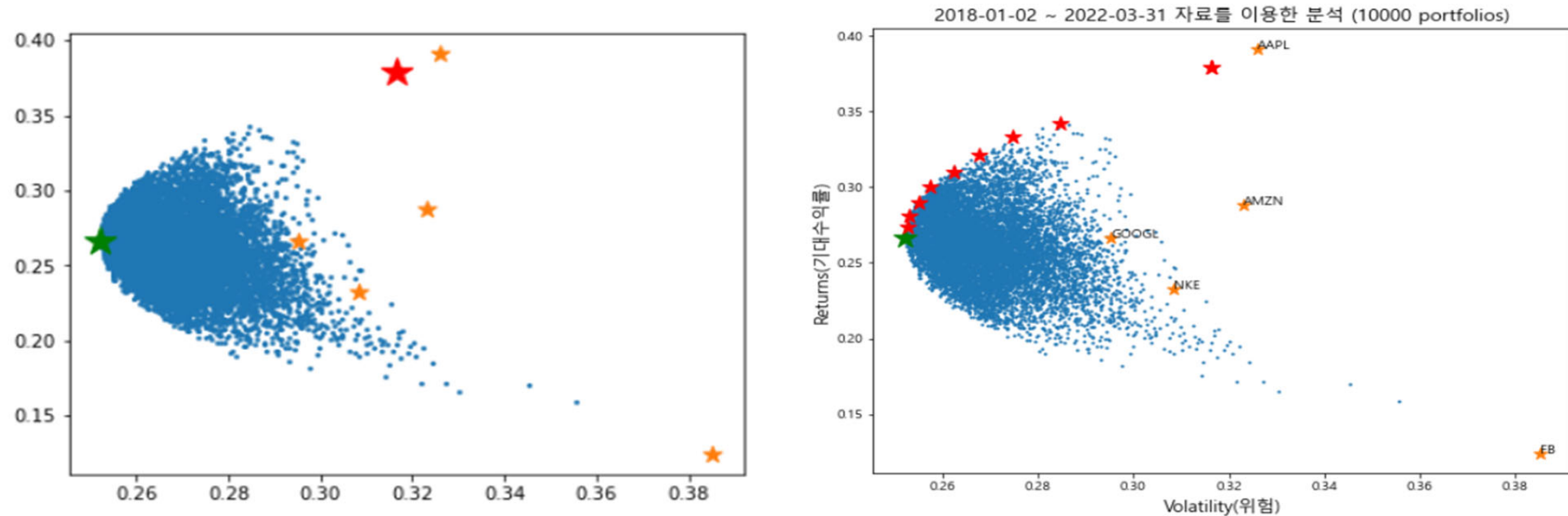
	AAPL weight	NKE weight	GOOGL weight	AMZN weight	FB weight	Returns	Volatility
0.27	0.147032	0.385836	0.295918	0.159694	0.011520	0.273498	0.252479
0.28	0.191389	0.359617	0.246188	0.191711	0.011094	0.280689	0.252843
0.29	0.267965	0.332203	0.249150	0.140213	0.010470	0.290173	0.254970
0.30	0.312056	0.294839	0.154601	0.237925	0.000578	0.300475	0.257307
0.31	0.410750	0.320251	0.065751	0.197376	0.005873	0.310329	0.262278
0.32	0.470119	0.217875	0.081424	0.220275	0.010306	0.321094	0.267593
0.33	0.568976	0.194525	0.059667	0.169059	0.007774	0.333505	0.274745
0.34	0.629971	0.108096	0.020431	0.213742	0.027760	0.342192	0.284516
0.35	0.921020	0.015108	0.006880	0.036943	0.020049	0.379028	0.316316
0.36	0.921020	0.015108	0.006880	0.036943	0.020049	0.379028	0.316316
0.37	0.921020	0.015108	0.006880	0.036943	0.020049	0.379028	0.316316

Expected return and variance of a portfolio

$$E(r_p) = \sum_{i=1}^{n} w_i E(r_i) \qquad \sigma_p^2 = \sum_{i=1}^{n} \sum_{j=1}^{n} w_i w_j Cov(r_i, r_j)$$

for loop

$$w = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix} \qquad V = \begin{bmatrix} \sigma_{1,1} & \cdots & \sigma_{1,n} \\ \vdots & \ddots & \vdots \\ \sigma_{n,1} & \cdots & \sigma_{n,n} \end{bmatrix} \qquad \sigma_p^2 = w^T V w$$

(n x 1)   (n x n)   (1 x n) (n x n) (n x 1)

numpy

Expected return and variance of a portfolio (pandas)

$$\sigma_p^2 = w^T V w$$

$$[w_1 \quad w_2 \quad \ldots \quad w_n] \begin{bmatrix} \sigma_{1,1} & \cdots & \sigma_{1,n} \\ \vdots & \ddots & \vdots \\ \sigma_{n,1} & \cdots & \sigma_{n,n} \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix}$$

cov_matrix.mul(weights, axis=0).mul(weights, axis=1).sum().sum()

dataframe   pandas

numpy

array   weights.T.dot(cov_matrix.dot(weights))

(1 x n)   (n x n)   (n x 1)

functions and methods

pd.DataFrame()

pd.concat()

df.pct_change()

df.cov()

df.mean()

df.std()

df.mul()

df.sum()

df.idxmin()

df.max()

np.random.rand()

np.sum()

np.dot()

np.sqrt()

np.arange()
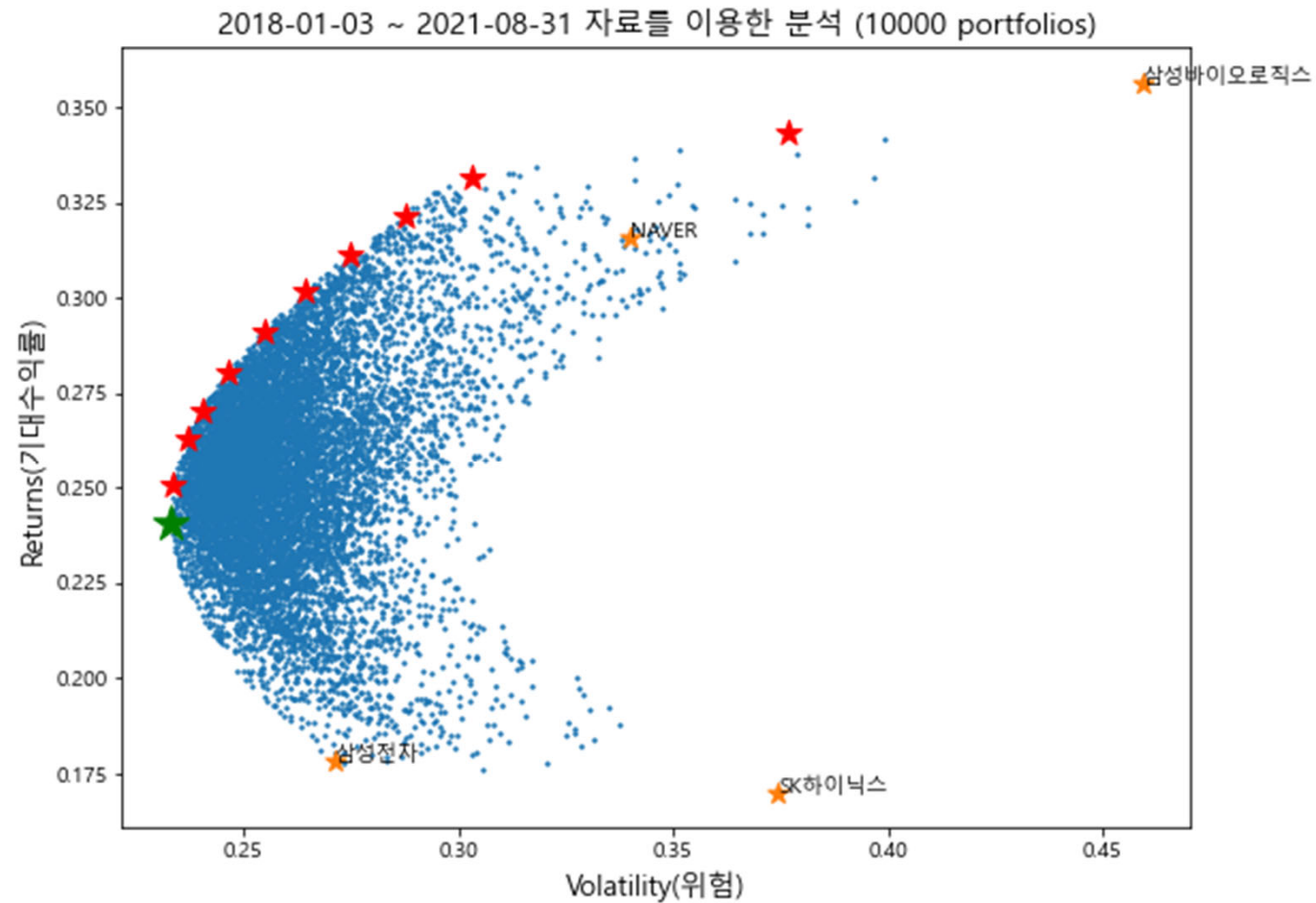
np.ceil()

plt.rc()

plt.subplots()

plt.scatter()

plt.xlabel()

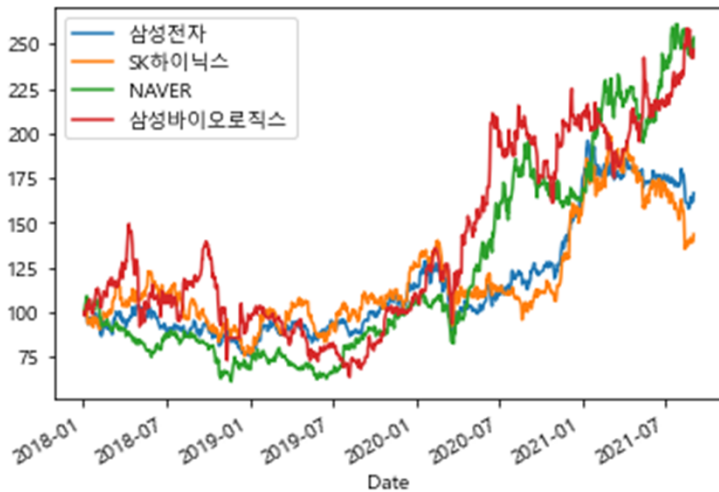plt.ylabel()

plt.title()

plt.text()

Korean stocks

2018-01-03 ~ 2021-08-31 자료를 이용한 분석 (10000 portfolios)

Stock prices

	Returns	Volatility
삼성전자	0.178423	0.271130
SK하이닉스	0.170023	0.374522
NAVER	0.315821	0.340016
삼성바이오로직스	0.356479	0.459365

	삼성전자	SK하이닉스	NAVER	삼성바이오로직스
삼성전자	1.000000	0.696140	0.321795	0.209871
SK하이닉스	0.696140	1.000000	0.277867	0.141706
NAVER	0.321795	0.277867	1.000000	0.196971
삼성바이오로직스	0.209871	0.141706	0.196971	1.000000