02. Multivariate Monte Carlo

August 2, 2021

[1]:
```
import pandas as pd
import numpy as np
```

[2]:
```
close_price_df = pd.read_csv("./sample_stock_close_price.csv", index_col=0,␣
↪parse_dates=True)
close_price_df.columns = ['Samsung Electronics', 'SK Hynix', 'KAKAO', 'NAVER',␣
↪'KODEX Inverse']
close_price_df.tail(3)
```

[2]:

	Samsung Electronics	SK Hynix	KAKAO	NAVER	KODEX Inverse
date					
2021-07-28	79200.0	114000.0	148000.0	442000.0	3765.0
2021-07-29	79000.0	114000.0	148500.0	439500.0	3755.0
2021-07-30	78500.0	112500.0	147000.0	433500.0	3805.0

0.0.1 Multivariate

$$X = \begin{bmatrix} X_1 \\ \vdots \\ X_n \end{bmatrix} \sim N\left(\begin{bmatrix} \mu_1 \\ \vdots \\ \mu_n \end{bmatrix}, \begin{bmatrix} \sigma_{1,1} & \dots & \sigma_{1,n} \\ \vdots & \ddots & \vdots \\ \sigma_{n,1} & \dots & \sigma_{n,n} \end{bmatrix} \right) \tag{1}$$

$$X \sim N(\mu, K) \tag{2}$$

$$K = L \cdot L^T \tag{3}$$

$$Z_{cor} = L \cdot \epsilon \tag{4}$$

$$\epsilon = Z_{ind} \times \sigma \tag{5}$$

$$Z_{ind} \sim N(0, 1) \tag{6}$$

$$Z_{cor} = L \cdot (Z_{ind} \times \sigma) \tag{7}$$

$$X \sim N(\mu, L \cdot (Z_{ind} \times \sigma)) \tag{8}$$

[3]:
```
returns_means = close_price_df.pct_change().mean()
returns_means
```

[3]:
```
Samsung Electronics    0.000700
SK Hynix               0.000828
KAKAO                  0.001075
NAVER                  0.000939
KODEX Inverse         -0.000254
dtype: float64
```

[4]:
```
returns_std = close_price_df.pct_change().std()
returns_std
```

[4]:
```
Samsung Electronics    0.017170
SK Hynix               0.024069
KAKAO                  0.023309
NAVER                  0.022415
KODEX Inverse          0.011115
dtype: float64
```

[5]:
```
rho_matrix = close_price_df.pct_change().corr()
rho_matrix
```

[5]:
```
                     Samsung Electronics  SK Hynix     KAKAO     NAVER  \
Samsung Electronics             1.000000  0.507238  0.149265  0.212039
SK Hynix                        0.507238  1.000000  0.177784  0.162638
KAKAO                           0.149265  0.177784  1.000000  0.319981
NAVER                           0.212039  0.162638  0.319981  1.000000
KODEX Inverse                  -0.732306 -0.571248 -0.294286 -0.338091

                     KODEX Inverse
Samsung Electronics      -0.732306
SK Hynix                 -0.571248
KAKAO                    -0.294286
NAVER                    -0.338091
KODEX Inverse             1.000000
```

[6]:
```
L = np.linalg.cholesky(rho_matrix)
pd.DataFrame(L, columns=close_price_df.columns)
```

[6]:
```
   Samsung Electronics  SK Hynix     KAKAO     NAVER  KODEX Inverse
0             1.000000  0.000000  0.000000  0.000000       0.000000
1             0.507238  0.861806  0.000000  0.000000       0.000000
2             0.149265  0.118439  0.981678  0.000000       0.000000
3             0.212039  0.063917  0.286000  0.932286       0.000000
```

4	-0.732306	-0.231832	-0.160461	-0.130973	0.605872

[7]:
```python
z_ind = np.random.normal(size=(100, len(close_price_df.columns)))
pd.DataFrame(z_ind, columns=close_price_df.columns)
```

[7]:

	Samsung Electronics	SK Hynix	KAKAO	NAVER	KODEX Inverse
0	-0.010418	-2.091732	-1.593299	-0.718171	0.458936
1	0.484347	0.444071	-0.009666	-1.077548	0.272307
2	0.224473	0.962848	0.303770	0.091169	0.321802
3	0.585634	3.560049	-0.417033	-0.818109	1.659349
4	-0.670369	0.026291	-1.262533	0.161998	-0.216747
..	…	…	…	…	…
95	0.262134	0.514455	0.343532	0.356085	1.357451
96	-1.751437	0.798450	-0.390593	-0.762740	0.820272
97	-0.828632	-0.393085	-0.322816	0.456986	0.010230
98	-0.042782	0.055115	1.421120	0.567286	0.426348
99	-0.119502	-1.585516	2.390145	-1.110405	-0.229470

[100 rows x 5 columns]

[8]:
```python
epsilon = pd.DataFrame(z_ind, columns=close_price_df.columns) * returns_std
epsilon
```

[8]:

	Samsung Electronics	SK Hynix	KAKAO	NAVER	KODEX Inverse
0	-0.000179	-0.050345	-0.037137	-0.016098	0.005101
1	0.008316	0.010688	-0.000225	-0.024153	0.003027
2	0.003854	0.023175	0.007080	0.002044	0.003577
3	0.010056	0.085686	-0.009720	-0.018338	0.018444
4	-0.011510	0.000633	-0.029428	0.003631	-0.002409
..	…	…	…	…	…
95	0.004501	0.012382	0.008007	0.007982	0.015089
96	-0.030073	0.019218	-0.009104	-0.017097	0.009118
97	-0.014228	-0.009461	-0.007524	0.010243	0.000114
98	-0.000735	0.001327	0.033124	0.012716	0.004739
99	-0.002052	-0.038161	0.055711	-0.024890	-0.002551

[100 rows x 5 columns]

[9]:
```python
z_cor = L.dot(epsilon.T).T
pd.DataFrame(z_cor, columns=close_price_df.columns)
```

[9]:

	Samsung Electronics	SK Hynix	KAKAO	NAVER	KODEX Inverse
0	-0.000179	-0.043479	-0.042447	-0.028885	0.022961
1	0.008316	0.013430	0.002286	-0.020136	-0.003535
2	0.003854	0.021927	0.010271	0.006229	-0.007432
3	0.010056	0.078945	0.002107	-0.012267	-0.012092
4	-0.011510	-0.005293	-0.030532	-0.007431	0.011069

```
..                  ...        ...        ...        ...            ...
95              0.004501  0.012954  0.009999  0.011477       0.000645
96             -0.030073  0.001308 -0.011150 -0.023691       0.026791
97             -0.014228 -0.015371 -0.010631  0.003776       0.012547
98             -0.000735  0.000771  0.032565  0.021257      -0.003879
99             -0.002052 -0.033929  0.049864 -0.010145       0.003125

[100 rows x 5 columns]
```

[10]:
```python
generated_X = returns_means + pd.DataFrame(z_cor, columns=close_price_df.
 ↪columns)
generated_X
```

[10]:
```
    Samsung Electronics  SK Hynix     KAKAO     NAVER  KODEX Inverse
0               0.000521 -0.042651 -0.041372 -0.027946       0.022707
1               0.009016  0.014258  0.003361 -0.019196      -0.003789
2               0.004554  0.022755  0.011346  0.007168      -0.007686
3               0.010755  0.079773  0.003182 -0.011328      -0.012346
4              -0.010811 -0.004465 -0.029457 -0.006492       0.010815
..                  ...        ...        ...        ...            ...
95              0.005201  0.013782  0.011074  0.012416       0.000391
96             -0.029373  0.002136 -0.010075 -0.022752       0.026537
97             -0.013528 -0.014542 -0.009556  0.004716       0.012293
98             -0.000035  0.001599  0.033640  0.022197      -0.004133
99             -0.001352 -0.033100  0.050939 -0.009206       0.002871

[100 rows x 5 columns]
```

[11]:
```python
generated_X.corr()
```

[11]:
```
                     Samsung Electronics  SK Hynix     KAKAO     NAVER  \
Samsung Electronics             1.000000  0.378364  0.222631  0.306941
SK Hynix                        0.378364  1.000000  0.099157  0.239514
KAKAO                           0.222631  0.099157  1.000000  0.499104
NAVER                           0.306941  0.239514  0.499104  1.000000
KODEX Inverse                  -0.790731 -0.620474 -0.475547 -0.601092

                     KODEX Inverse
Samsung Electronics      -0.790731
SK Hynix                 -0.620474
KAKAO                    -0.475547
NAVER                    -0.601092
KODEX Inverse             1.000000
```