python3 datageneration.py "Peilin Luo" "./"

Final Solution

Peilin Luo (pl212)

Note: I used my risk management library in the solution. The latest version of the library has been uploaded to the Repo, along with the instructions for use.

Q1

In problem1.csv, we have 3 price variables with missing values. We need to calculate the log return of these 3 assets, compute the covariance matrix with pairwise method, and fix the non-PSD covariance matrix with ‘near_psd’ method.

The log return is:

	price1	price2	price3
2023-04-13	-0.000630	-0.000902	-0.005475
2023-04-14	0.000194	0.000219	0.002367
2023-04-15	0.000166	-0.001653	-0.011529
2023-04-16	0.000043	0.002504	0.015642
2023-04-17	NaN	-0.002004	-0.015058
2023-04-18	NaN	0.000956	0.009691
2023-04-19	0.000109	0.000683	0.005197
2023-04-20	-0.000393	0.000094	-0.002241
2023-04-21	0.000546	0.001714	0.011144
2023-04-22	-0.000484	-0.003213	-0.019639
2023-04-23	0.000086	0.001718	0.010433
2023-04-24	0.000165	-0.001458	-0.010662
2023-04-25	-0.000156	0.000114	0.002607
2023-04-26	0.000819	0.002998	0.015875
2023-04-27	-0.001084	-0.002590	-0.011714
2023-04-28	0.000178	0.000775	0.003298
2023-04-29	NaN	NaN	-0.004857
2023-04-30	NaN	NaN	NaN
2023-05-01	0.000130	NaN	NaN

Its pairwise covariance matrix is:

	Price1	Price2	Price3
Price1	2.221428e-07	6.380692e-07	0.000003
Price2	6.380692e-07	3.327751e-06	0.000020
Price3	3.453138e-06	2.019423e-05	0.000120

The eigenvalue of the covariance matrix is:

-7.366861e-08
1.399980e-07
1.239406e-04

The first eigenvalue is significantly negative (below $-1e^{-8}$). So the matrix is non-PSD. We then fixed the matrix to PSD with the 'near_psd' method. The fixed covariance matrix is:

2.221428e-07	6.347312e-07	0.000003
6.347312e-07	3.327751e-06	0.000020
3.447019e-06	1.991895e-05	0.000120

The eigenvalue of the fixed matrix is:

5.460354e-21
1.558270e-07
1.238511e-04

They are all above 0, so the fixed covariance matrix is PSD.

When we use financial price data worldwide, there might be missing values since the exchanges of different countries operate at different times. Not all markets are open at the same time on the same days. A holiday in one market is not necessarily a holiday in another, even in the same country. It is very hard to match.

Q2

In problem 2, we need to calculate the value and Greeks of one European call option. The results are in the table below.

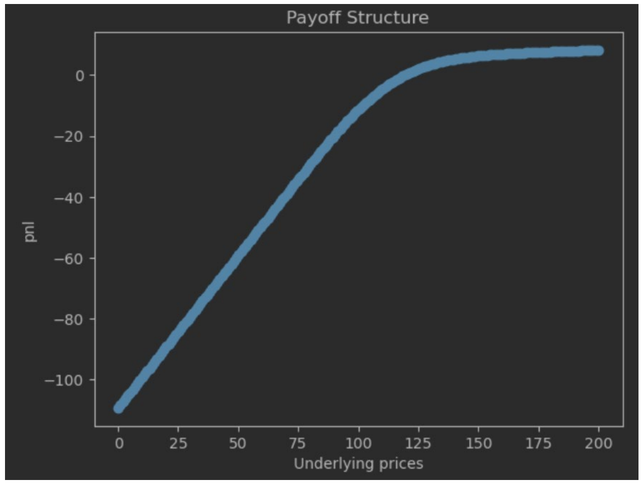
Value	Delta	Gamma	Vega	Rho
10.14	0.5972	0.0178	33.93	35.50

Now we set up a portfolio by buying one share and selling one call. Then assuming that the stock daily return obeys the normal distribution, $N(0, 0.0144^2)$, we need to simulate the portfolio's pnl one day ahead and calculate the VaR and ES.

Using Monte Carlo simulation, the VaR and the ES of the portfolio are 1.22$ and 1.56$ respectively. looking at the portfolio's pnl distribution, the pnl mean is negative, along with a negative skewness and an excessive kurtosis, this portfolio is not worth investing in and pretty risky.

Pnl mean	Pnl std	Pnl skewness	Pnl kurtosis
-0.024	0.698	-0.242	0.099

We can also look at the payoff chart of the portfolio below. This is a covered call portfolio and the payoff shape is the same as shorting a put, according to put call parity.



Q3

In problem 3, we need to compute the Max Sharpe Ratio portfolio and the Risk Parity portfolio with the expected return and covariance of 3 assets. Then we compared these 2 portfolios and explain the difference.

For the Max Sharpe Ratio portfolio, the weights are:

Asset 1	Asset 2	Asset 3
0.301223	0.350770	0.348007

For the Risk Parity portfolio, the weights are:

Asset 1	Asset 2	Asset 3
0.301354	0.350835	0.347811

Notice that the super-efficient portfolio is pretty much like the risk parity portfolio because the correlations between assets are similar (around 0.012), and their Sharpe ratios are also very similar (see table below). If all the correlations are equal, the Risk Parity portfolio is the inverse volatility portfolio. If all correlations are equal and all Sharpe Ratios are equal, then the Risk Parity portfolio is also the Maximum Sharpe Ratio, portfolio.

	Asset 1	Asset 2	Asset 3
Sharpe Ratio	1.885016	2.194522	2.175609

Q4

In problem 4, we have the returns of 3 assets and the starting weight of them. We need to compute the weights through time, and then the ex-post return attribution and risk attribution of the portfolio.

Here are the weights through time:

	asset1	asset2	asset3
0	0.490952	0.082704	0.426344
1	0.494777	0.076275	0.428948
2	0.505588	0.076416	0.417996
3	0.505268	0.077743	0.416989
4	0.498896	0.086448	0.414656
5	0.489767	0.084329	0.425904
6	0.486626	0.080457	0.432918
7	0.519955	0.075949	0.404096
8	0.485486	0.077706	0.436808
9	0.468067	0.080754	0.451179
10	0.435867	0.087532	0.476602
11	0.449715	0.088222	0.462063
12	0.461237	0.092196	0.446567
13	0.454174	0.086602	0.459224
14	0.446349	0.080572	0.473079
15	0.451625	0.075224	0.473151
16	0.496436	0.067592	0.435971
17	0.494874	0.071098	0.434028
18	0.474674	0.075746	0.449580
19	0.501675	0.070505	0.427819

To compute the ex-post return attribution, we use Carino's k to make returns summable. To compute the ex-post risk attribution, we calculated the regression coefficient of the weighted returns regressed on the portfolio return and multiplied it by the portfolio standard deviation. The results are in the table below.

	Asset1	Asset2	Asset3	Portfolio
TotalReturn	0.080428	-0.062929	-0.037279	0.018388
Return Attribution	0.037525	-0.005434	-0.013702	0.018388
Vol Attribution	0.026484	0.000376	0.007973	0.034832

Q5

In problem 5, we have 4 assets, and their returns all obey the generalized T distribution. To calculate the 5% VaR for each asset, we first fit each asset's return to T distribution, then select the 5^{th} percentile. The results are in the table below.

1	2	3	4
0.09%	0.06%	0.05%	0.07%

To calculate the VaR for a portfolio of asset 1&2, and a portfolio of asset 3&4, we used Gaussian Copula. To be specific, we transformed the original returns to U matrix with the corresponding CDF, then computed the spearman correlation of the U matrix. We simulated from the spearman correlation matrix with Monte Carlo, and applied the CDF of standard normal on them to get a simulated U. Then we transformed U back into X with the corresponding quantile functions. The results are in the table below.

Asset 1&2	Asset 3&4	All asset
14.5%	11.03%	24.25%