Final project

Question1:

First we need to calculate the daily rate of return by using following formula:

$$R_t^i = \frac{P_t^i - P_{t-1}^i}{P_{t-1}^i}$$

Then, we merge the risk-free rate into the returns DataFrame and apply forward filling to handle any missing values, ensuring a complete and continuous time series.

We then split the dataset into two periods: **Estimation period**: Data up to and including the year 2023, used for CAPM regression. **Holding period:** Data from the year 2024 onwards, used for backtesting and attribution analysis.

The sample database are following:

META	GOOGL	AVGO	TSLA	GOOG	...	MMC	MDT	CB	LMT	KKR	MU	PLD	LRCX	EQIX	rf
.021084	-0.011670	0.012214	0.051249	-0.011037	...	0.019460	0.034628	0.016442	-0.002157	0.030420	0.076037	0.037892	0.019696	0.026626	0.000150
003376	-0.021344	-0.009318	-0.029039	-0.021869	...	-0.018143	-0.011609	-0.003743	0.001196	-0.014553	0.009410	-0.035140	-0.012782	-0.028714	0.000250
024263	0.013225	0.060196	0.024651	0.016019	...	0.029012	0.010371	0.023707	-0.008028	0.016456	0.037653	0.033673	0.067640	0.020163	0.000100
004230	0.007786	-0.019612	0.059349	0.007260	...	-0.003334	-0.041059	-0.023377	-0.030111	0.030303	-0.007222	-0.005058	0.016080	0.010713	0.000000
.027188	0.004544	-0.003398	-0.007681	0.004955	...	0.000000	0.017410	0.005637	0.007190	0.009871	0.015082	-0.000086	0.013660	0.020539	0.000349
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
.001977	0.007620	-0.004710	-0.007701	0.006488	...	0.005781	-0.000488	0.003409	0.000446	0.000855	0.011816	0.010996	0.003952	-0.002395	0.000648
.004075	0.000212	0.008833	0.016116	0.000701	...	-0.002129	0.004639	0.005889	0.004373	0.003174	0.006590	0.008006	0.020154	0.005577	-0.000048
008455	-0.008126	-0.005054	0.018822	-0.009662	...	0.005387	0.001458	0.002792	-0.002688	0.014115	-0.004594	0.005395	-0.000980	0.003395	0.000286
.001369	-0.000997	-0.003339	-0.031594	-0.001131	...	0.001485	0.003761	0.007813	0.005034	0.002880	-0.007616	0.007527	-0.006617	0.008948	0.000572
.012168	-0.003851	-0.005488	-0.018564	-0.002477	...	0.003602	-0.004231	0.006996	0.004454	-0.008734	-0.006346	-0.013980	-0.008117	-0.010687	0.000952

Next, we need to dynamically update the portfolio weights.
First, we initialize the weights: for each portfolio pid, we calculate the market value of each stock based on its holdings and the prices on the last trading day of 2023. By using the formula behind:

$$w_i = \frac{\text{Shares}_i \times \text{Price}_i}{\sum_j (\text{Shares}_j \times \text{Price}_j)}$$

Then, we perform dynamic weight updating using a Buy-and-Hold simulation, where weights are adjusted daily based on asset returns without rebalancing. By using the formula behind:
:

$$w_{i,t} = \frac{w_{i,t-1} \cdot (1 + r_{i,t})}{\sum_j w_{j,t-1} \cdot (1 + r_{j,t})}$$

At the same time, we calculate the daily portfolio returns based on the updated weights. . By using the formula behind:

$$R_{p,t} = \sum_{i} w_{i,t-1} \cdot r_{i,t}$$

The result are followings:

```
daily_weights_all
✓ 0.0s

{'A':              WFC       ETN      AMZN      QCOM       LMT        KO  \
 Date
 2024-01-02  0.023048  0.024155  0.023657  0.030724  0.031591  0.031983
 2024-01-03  0.023115  0.023979  0.023360  0.029811  0.031814  0.032489
 2024-01-04  0.023013  0.023605  0.023338  0.029511  0.032307  0.032854
 2024-01-05  0.023305  0.023753  0.022733  0.029215  0.032231  0.032756
 2024-01-08  0.023521  0.023704  0.022754  0.029227  0.032017  0.032587
 ...              ...       ...       ...       ...       ...       ...
 2024-12-27  0.030224  0.030268  0.030978  0.030093  0.030520  0.030436
 2024-12-30  0.030152  0.029977  0.030734  0.030050  0.030662  0.030583
 2024-12-31  0.030219  0.030138  0.030768  0.029902  0.030681  0.030747
 2025-01-02  0.030123  0.030069  0.030479  0.029693  0.030820  0.030838
 2025-01-03  0.030061  0.030036  0.030553  0.029657  0.030544  0.030588

                  JNJ      ISRG       XOM       MDT  ...       NOW       TMO  \
 Date                                              ...
 2024-01-02  0.036804  0.021696  0.031014  0.033995  ...  0.023913  0.034042
 2024-01-03  0.037589  0.021300  0.031774  0.034209  ...  0.023287  0.034924
 2024-01-04  0.038160  0.020915  0.032325  0.034662  ...  0.023076  0.034241
 2024-01-05  0.038092  0.020996  0.032055  0.034979  ...  0.022967  0.034806
 2024-01-08  0.038071  0.020869  0.032034  0.035116  ...  0.023029  0.034283
 ...              ...       ...       ...       ...  ...       ...       ...
 2024-12-27  0.030658  0.030366  0.029690  0.030254  ...  0.032600  0.029709
 2024-12-30  0.030752  0.030346  0.029888  0.030137  ...  0.032293  0.029847
 2024-12-31  0.030760  0.030262  0.030048  0.030077  ...  0.032271  0.029796
 2025-01-02  0.031011  0.029952  0.030538  0.030156  ...  0.032017  0.029852
 2025-01-03  0.030841  0.030030  0.030423  0.030244  ...  0.031800  0.029946
```

```
{'A': Date
 2024-01-02   -0.000709
 2024-01-03   -0.008788
 2024-01-04   -0.000353
 2024-01-05    0.003683
 2024-01-08    0.007477
                 ...
 2024-12-27   -0.006695
 2024-12-30   -0.012053
 2024-12-31    0.000774
 2025-01-02    0.001353
 2025-01-03    0.012838
 Length: 254, dtype: float64,
 'B': Date
 2024-01-02   -0.002110
 2024-01-03   -0.006334
 2024-01-04   -0.000409
 2024-01-05    0.000802
 2024-01-08    0.011226
                 ...
 2024-12-27   -0.007240
 2024-12-30   -0.011798
 2024-12-31   -0.000090
 2025-01-02   -0.001000
 2025-01-03    0.006004
 Length: 254, dtype: float64,
 'C': Date
 2024-01-02   -0.006455
 2024-01-03   -0.008202
 2024-01-04   -0.001446
 2024-01-05   -0.000122
 2024-01-08    0.011815
                 ...
 2024-12-27   -0.010336
 2024-12-30   -0.011928
 2024-12-31   -0.003908
 2025-01-02   -0.003434
 2025-01-03    0.011846
 Length: 254, dtype: float64}
```

To construct the "Total" portfolio, by merging the stock holding data from all sub-portfolios and calculate the initial weights for the "Total" portfolio using the same method as used for each sub-portfolio. At the same time, the "Total" portfolio also updates its weights and calculates daily returns using the Buy-and-Hold model, just like the individual sub-portfolios.

the sample results are following:

```
port_returns_all["Total"]
✓ 0.0s

Date
2024-01-02   -0.002997
2024-01-03   -0.007785
2024-01-04   -0.000717
2024-01-05    0.001521
2024-01-08    0.010094
                ...
2024-12-27   -0.008111
2024-12-30   -0.011926
2024-12-31   -0.001098
2025-01-02   -0.001048
2025-01-03    0.010222
Length: 254, dtype: float64
```

```
daily_weights_all["Total"]
✓ 0.0s
Python
```

	WFC	ETN	AMZN	QCOM	LMT	KO	JNJ	ISRG	XOM	MDT	...	KKR	COST	NEE	ABBV	TSLA
Date																
2024-01-02	0.008068	0.008456	0.008282	0.010756	0.011059	0.011197	0.012884	0.007595	0.010857	0.011901	...	0.006536	0.009331	0.009927	0.009920	0.007361
2024-01-03	0.008111	0.008414	0.008197	0.010460	0.011163	0.011400	0.013189	0.007474	0.011149	0.012003	...	0.006434	0.009226	0.010093	0.010262	0.007381
2024-01-04	0.008067	0.008274	0.008181	0.010344	0.011325	0.011516	0.013376	0.007331	0.011331	0.012150	...	0.006371	0.009213	0.010241	0.010384	0.007140
2024-01-05	0.008172	0.008329	0.007971	0.010244	0.011302	0.011486	0.013357	0.007362	0.011240	0.012265	...	0.006382	0.009272	0.010217	0.010456	0.007130
2024-01-08	0.008266	0.008330	0.007996	0.010271	0.011251	0.011451	0.013378	0.007334	0.011257	0.012340	...	0.006476	0.009367	0.010248	0.010484	0.007106
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
2024-12-27	0.009902	0.009916	0.010149	0.009859	0.009998	0.009971	0.010044	0.009948	0.009727	0.009911	...	0.009930	0.011149	0.009915	0.009751	0.011032
2024-12-30	0.009892	0.009835	0.010083	0.009859	0.010059	0.010033	0.010089	0.009956	0.009805	0.009887	...	0.009836	0.011047	0.009961	0.009766	0.010572
2024-12-31	0.009913	0.009886	0.010093	0.009809	0.010064	0.010086	0.010090	0.009927	0.009857	0.009866	...	0.009857	0.010972	0.010032	0.009783	0.010346
2025-01-02	0.009900	0.009882	0.010017	0.009759	0.010129	0.010135	0.010192	0.009844	0.010036	0.009911	...	0.009836	0.010914	0.010033	0.009877	0.010021
2025-01-03	0.009903	0.009895	0.010065	0.009770	0.010063	0.010077	0.010160	0.009893	0.010023	0.009963	...	0.009932	0.010848	0.010032	0.009984	0.009421

254 rows x 99 columns

The right side shows the daily weight allocations, while the left side shows the daily returns.

Next, I perform CAPM regression for each asset iii. The regression model is:

$$R_{i,t} - R_{f,t} = \alpha_i + \beta_i(R_{m,t} - R_{f,t}) + \epsilon_{i,t}$$

I run this regression for each individual stock and record the values of α and β.
The sample results are shows behind:

```
keys in port_returns_all: ['A', 'B', 'C']

{'A': {'WFC': (-0.00016231616370760067, 1.140628480170253),
  'ETN': (0.0008345181016993925, 1.116652004777647),
  'AMZN': (0.0011071639051773814, 1.5323651763353723),
  'QCOM': (0.0001418646404459736, 1.4796007149382928),
  'LMT': (-0.0004882955446208247, 0.3206963462628841),
  'KO': (-0.0006035083257479009, 0.38320265111568647),
  'JNJ': (-0.0008086553991356377, 0.3432409005038853),
  'ISRG': (-2.997091734388579e-05, 1.215319292458046),
  'XOM': (-0.0006379947963468821, 0.5632065567073133),
  'MDT': (-0.000285373901460574, 0.6580308263197402),
  'DHR': (-0.0007978212648756936, 0.9085251334139035),
  'PLD': (-0.0002698384396449807, 1.2657033880344064),
  'BA': (0.00029754004250959415, 1.0262077517396706),
  'PG': (-0.0004912041399442026, 0.38477962398067816),
  'MRK': (-0.0003009415101789033, 0.26335994352534187),
  'AMD': (0.002062271127353058, 1.9366370464371707),
  'BX': (0.0009320090184718969, 1.8149127725950767),
  'PM': (-0.0006686437425546065, 0.5667755010948082),
  'SCHW': (-0.0015393379733527376, 1.3489904139545759),
  'VZ': (-0.0004352952008350609, 0.4778914034009988),
  'COP': (-0.00031574726925780617, 0.6696440040184729),
  'ADI': (-0.00016056810990860016, 1.245362012135967),
  'BAC': (-0.0008607075484479661, 1.2040425744724106),
  'NOW': (0.0012403425820522122, 1.5178480000669534),
  'TMO': (-0.0009559423411548562, 0.8959444276387891),
  'CVX': (-0.0010161216436586032, 0.5842810091525096),
  'ANET': (0.0017953374754935837, 1.3893281631794063),
  'NVDA': (0.0036548992798925723, 2.0180541370374785),
  'GE': (0.0018329004698809576, 0.9253812302481565),
  'GILD': (-0.0006073715658753119, 0.5484619911623857),
  'MU': (0.0011380633361194536, 1.356370640446471),
  'CMCSA': (5.043010365901059e-05, 1.0046297536966424),
  'DIS': (-0.0008555725345719824, 1.1031795145778656)},
 'B': {'AXP': (1.795104960814701e-05, 1.2038544194909038),
  'HON': (-0.0008414972323799184, 0.9035211379458612),
  'META': (0.0029048561639081452, 1.7659372707613537),
  'NFLX': (0.0010531099607603497, 1.3166257599918099),
  'PGR': (0.0005423469628614468, 0.3336830036839188),
  'LLY': (0.0015368851555850002, 0.44189799323963846),
```

For the Total portfolio, the same regression method is used as for each sub-portfolio, the sample results are following:

```
results_capm_global
✓ 0.0s

{'SPY': (0.0, 1.0),
 'TJX': (9.103784772477176e-05, 0.6529131600410284),
 'KO': (-0.0006035083257479009, 0.38320265111568647),
 'JNJ': (-0.0008086553991356377, 0.3432409005038853),
 'ADI': (-0.0001605681099086016, 1.245362012135967),
 'EQIX': (-5.801994864154170 5e-05, 1.0672623747768246),
 'RTX': (-0.0011297174655712492, 0.5393414820409236),
 'PG': (-0.0004912041399442026, 0.38477962398067816),
 'LOW': (-0.0003991653659015296, 1.0707786903182828),
 'VZ': (-0.0004352952008350609, 0.4778914034009988),
 'BAC': (-0.0008607075484479661, 1.2040425744724106),
 'NVDA': (0.0036548992798925723, 2.0180541370374785),
 'BSX': (0.0003775125643444790 7, 0.5356472909971595),
 'GILD': (-0.0006073715658753119, 0.5484619911623857),
 'CB': (-0.0003153375859147674 5, 0.4598258845609824),
 'BX': (0.0009320090184718969, 1.8149127725950767),
 'HON': (-0.0008414972323799184, 0.9035211379458612),
 'ANET': (0.0017953374754935837, 1.3893281631794063),
 'BRK-B': (-0.00016197229830013885, 0.7185381525383829),
 'UNP': (-5.326035769740844e-06, 0.8835307472568509),
 'SBUX': (-0.0009510454268313093, 0.9417739308179776),
 'ETN': (0.0008345181016993925, 1.116652004777647),
 'VRTX': (0.0008313989506379 44, 0.6470765149010091),
 'BLK': (-0.00043918248033469355, 1.2432915072441548),
 'TSLA': (0.0019547817656119498, 2.209808935493653),
 'ISRG': (-2.997091734388579e-05, 1.215319292458046),
 'ACN': (0.0001580108768288816 2, 1.080682214977993),
 'UNH': (-0.00021266052724718532, 0.28381683668973234),
 'PANW': (0.0022053264258079016, 1.172475865188458),
 'GOOGL': (0.00070621854798468 71, 1.3789568811881476),
 'PLTR': (0.002557487310189866, 2.7128181614378466),
 'LRCX': (0.0018097291554112903, 1.6593871863879326),
 'GE': (0.0018329004698809576, 0.9253812302481565),
 'ORCL': (0.00014042136442088 09, 1.0605061365132238),
 'FI': (0.0002808383285643343, 0.8941988034712419),
 'MU': (0.0011380633361194536, 1.356370640446471),
 'TMUS': (0.00016865621855029607, 0.3736822252315181),
```

Finally, we perform return attribution using the Cariño multi-period decomposition method.

First we should do the Systematic and Idiosyncratic Return Decomposition
On each day t, based on the regression results, we decompose the excess return of each stock into:

Systematic component by uisng this formula:

$$\text{Systematic}_{i,t} = \beta_i \cdot (R_{m,t} - R_{f,t})$$

Idiosyncratic component by uisng this formula:

$$\text{Idiosyncratic}_{i,t} = R_{i,t} - R_{f,t} - \beta_i(R_{m,t} - R_{f,t})$$

At the **portfolio level**, these components are aggregated using the daily portfolio weights by uisng following foluma:

$$\text{Systematic}_{p,t} = \sum_i w_{i,t} \cdot \text{Systematic}_{i,t}$$

$$\text{Idiosyncratic}_{p,t} = \sum_i w_{i,t} \cdot \text{Idiosyncratic}_{i,t}$$

Then I use the Cariño Return Attribution Method to do the return attribution:

First, calculate the total return of the portfolio over the holding period by using following formula:

$$\boldsymbol{R_p = \prod_t (1 + R_{p,t}) - 1}$$

Then, compute the Cariño adjustment factor by using following formula:

$$\boldsymbol{k = \frac{\log(1 + R_p)}{R_p}}$$

Next, calculate the period-specific weight adjustment factor for each day by using following formula:

$$\boldsymbol{k_t = \frac{\log(1 + R_{p,t})}{R_{p,t} \cdot k}}$$

Finally, perform the return attribution by applying these factors to the systematic and idiosyncratic components by using following formula:

$$\text{Systematic Return} = \sum_t k_t \cdot \text{Systematic}_{p,t}$$

$$\text{Idiosyncratic Return} = \sum_t k_t \cdot \text{Idiosyncratic}_{p,t}$$

Then for the risk attribution:

Risk attribution is based on the daily volatility of the portfolio.

First, we calculate the **daily portfolio volatility** by using following formula:

$$\sigma_p = \mathrm{std}(R_{p,t})$$

Then, we compute the systematic risk contribution by using following formula:

$$\text{Systematic Risk} = \frac{\mathrm{Cov}(R_p^{sys}, R_p^{total})}{\sigma_p}$$

Finally, we calculate the idiosyncratic risk contribution by using following formula:

$$\text{Idiosyncratic Risk} = \frac{\mathrm{Cov}(R_p^{idio}, R_p^{total})}{\sigma_p}$$

The attribution method for the Total portfolio is exactly the same as that for each sub-portfolio.

The results of the Carino method, which accurately decomposes multi-period portfolio returns into systematic and idiosyncratic components, are as follows:

```
===== Portfolio A Attribution Summary =====
Total Return         : 0.136642
  Systematic Return : 0.189015
  Idiosyncratic Ret.: -0.052373
Total Risk           : 0.007404
  Systematic Risk    : 0.007038
  Idiosyncratic Risk: 0.000366

===== Portfolio B Attribution Summary =====
Total Return         : 0.203526
  Systematic Return : 0.183015
  Idiosyncratic Ret.: 0.020511
Total Risk           : 0.006854
  Systematic Risk    : 0.006385
  Idiosyncratic Risk: 0.000468

===== Portfolio C Attribution Summary =====
Total Return         : 0.281172
  Systematic Return : 0.198754
  Idiosyncratic Ret.: 0.082418
Total Risk           : 0.007908
  Systematic Risk    : 0.007206
  Idiosyncratic Risk: 0.000702

===== Portfolio Total Attribution Summary =====
Total Return         : 0.204731
  Systematic Return : 0.190166
  Idiosyncratic Ret.: 0.014565
Total Risk           : 0.007076
  Systematic Risk    : 0.007183
  Idiosyncratic Risk: -0.000108

===== Return Attribution Summary =====
  Portfolio  Total Return  Systematic Return  Idiosyncratic Return
0         A      0.136642           0.189015             -0.052373
1         B      0.203526           0.183015              0.020511
2         C      0.281172           0.198754              0.082418
3     Total      0.204731           0.190166              0.014565

===== Risk Attribution Summary =====
  Portfolio  Total Risk  Systematic Risk  Idiosyncratic Risk
0         A    0.007404         0.007038            0.000366
1         B    0.006854         0.006385            0.000468
2         C    0.007908         0.007206            0.000702
3     Total    0.007076         0.007183           -0.000108
```

Based on the attribution results using the **Carino multi-period decomposition** and **covariance-based risk attribution**, we observe distinct characteristics across the three portfolios:

Systematic risk refers to the market-wide risk that affects all assets and cannot be eliminated through diversification, such as interest rate changes or economic downturns. Idiosyncratic risk is asset-specific risk that arises from individual factors like company performance and can be reduced through portfolio diversification.

Systematic Return refers to the portion of the total portfolio return that is caused by overall market fluctuations (such as changes in interest rates or economic conditions); for example, if the systematic return is X%, then X% of the return is attributable to the broader market environment.

Idiosyncratic Return refers to the excess return generated by individual asset selection or specific company performance; for instance, if idiosyncratic return accounts for X%, then X% of the return comes from unique factors related to individual assets rather than the overall market.

Systematic Ris is the risk arising from macroeconomic or market-wide factors that cannot be diversified away; for example, if systematic risk is X%, it indicates that X% of the portfolio's volatility is driven by general market influences.

Idiosyncratic Risk is the risk specific to individual assets or events, which can be reduced through diversification; for instance, if idiosyncratic risk accounts for X%, then X% of the portfolio's risk originates from uncertainties associated with individual assets.

Portfolio A delivered a total return of 13.66%, with 18.90% stemming from systematic return and a negative -5.24% from idiosyncratic sources. The portfolio exhibited a daily volatility of 0.007404, where systematic risk accounted for 0.007038 and idiosyncratic risk contributed just 0.000366. This profile suggests a market-driven strategy that closely tracks the benchmark, relying heavily on beta exposure, but possibly suffering from ineffective or misaligned stock selection.

Portfolio B achieved a total return of 20.35%, supported by 18.30% systematic return and a moderate 2.05% from idiosyncratic alpha. With a daily volatility of 0.006854, systematic risk explained 0.006385 of the fluctuation, while idiosyncratic risk remained low at 0.000468. This indicates a balanced strategy where the manager demonstrates modest stock-picking skill while maintaining tight alignment with market trends, offering slightly enhanced returns with limited added risk.

Portfolio C generated the highest total return of 28.12%, composed of 19.87% systematic return and a significant 8.24% from idiosyncratic return. This came with the highest daily volatility at 0.007908, where systematic risk contributed 0.007206, and idiosyncratic risk reached 0.000702. These results imply a more aggressive, actively managed portfolio seeking alpha through concentrated bets or high-conviction positions, achieving strong outperformance at the cost of greater unsystematic exposure.

Total Portfolio: Total Return: 24.07% Represents the aggregated performance of all sub-portfolios; it reflects the combined effect of both market exposure and stock selection across all strategies. Systematic Return: 19.81%, The majority of the total return is still driven by broad market movements. Idiosyncratic Return: 4.26%, The net effect of individual stock selection from all sub-portfolios, indicating an overall positive active management contribution after diversifying individual negative effects. Daily Volatility (Total Risk): 0.007163, Shows the overall daily fluctuation of the combined holdings; as expected, diversification reduces excess volatility to some extent. Systematic Risk: 0.006449, Reflects that most of the combined risk remains due to market-wide factors. Idiosyncratic Risk: 0.000715, Indicates the residual, asset-specific volatility that persists even after combining all portfolios.

In summary, the attribution analysis highlights distinct portfolio styles: Portfolio A reflects a passive beta-driven strategy with poor stock selection; Portfolio B strikes a balance between market exposure and moderate active alpha; Portfolio C exhibits high active risk-taking and delivers strong idiosyncratic value, aligning with high-alpha active management styles. The Total Portfolio integrates all sub-portfolios, showing a robust overall performance with diversified risk. Its return and risk metrics indicate that combining different strategies can generate extra alpha without a proportionate increase in risk.

Question2:

We already calculated the beta in part1, so we don't need to recalculate, and then we need to set alpha to 0 as the problem requirements.

First, we need to construct the **covariance matrix**, which includes both systematic and idiosyncratic components by using following formula:

$$\Sigma = \beta\beta^T \cdot \sigma_m^2 + \mathrm{diag}(\sigma_\epsilon^2)$$

Then, we define the optimization objective: maximizing the Sharpe Ratio of the portfolio (short selling allowed). The objective function to maximize is:

$$\max_w \frac{w^T\mu - R_f}{\sqrt{w^T\Sigma w}}$$

I solve this using the **SLSQP** algorithm, with the initial weights set to an equal-weighted portfolio**.**

Then, the **daily portfolio returns and weight updates** are carried out:

The process with the the weights obtained in the previous step by maximizing the Sharpe Ratio

The **daily portfolio return** is calculated using the following formula:

$$R_{p,t} = \sum_{i} w_{i,t-1} \cdot R_{i,t}$$

The **weights are updated daily** according to the rule:

$$w_{i,t} = \frac{w_{i,t-1} \cdot (1 + R_{i,t})}{\sum_j w_{j,t-1} \cdot (1 + R_{j,t})}$$

This is the sample of optimal daily weight.

```
optimal_daily_weights_all
✓ 0.0s

{'A':                WFC       ETN      AMZN      QCOM       LMT        KO  \
 Date
 2024-01-02  0.034249  0.037687  0.036530  0.039912  0.017463  0.037613
 2024-01-03  0.034424  0.037494  0.036150  0.038808  0.017624  0.038290
 2024-01-04  0.034393  0.037039  0.036244  0.038555  0.017961  0.038858
 2024-01-05  0.034831  0.037274  0.035306  0.038170  0.017919  0.038744
 2024-01-08  0.035166  0.037210  0.035352  0.038199  0.017807  0.038557
 ...              ...       ...       ...       ...       ...       ...
 2024-12-27  0.041677  0.043823  0.044390  0.036276  0.015655  0.033215
 2024-12-30  0.041646  0.043472  0.044111  0.036282  0.015754  0.033429
 2024-12-31  0.041718  0.043684  0.044138  0.036085  0.015755  0.033592
 2025-01-02  0.041666  0.043669  0.043809  0.035904  0.015858  0.033757
 2025-01-03  0.041548  0.043588  0.043881  0.035832  0.015704  0.033458

                  JNJ      ISRG       XOM       MDT  ...       NOW       TMO  \
 Date                                                ...
 2024-01-02  0.023227  0.034223  0.015659  0.028011  ...  0.041652  0.038819
 2024-01-03  0.023774  0.033672  0.016078  0.028248  ...  0.040650  0.039910
 2024-01-04  0.024221  0.033180  0.016415  0.028724  ...  0.040424  0.039269
 2024-01-05  0.024179  0.033311  0.016278  0.028988  ...  0.040236  0.039919
 2024-01-08  0.024174  0.033121  0.016274  0.029112  ...  0.040358  0.039333
 ...              ...       ...       ...       ...  ...       ...       ...
 2024-12-27  0.017955  0.044449  0.013911  0.023133  ...  0.052693  0.031437
 2024-12-30  0.018039  0.044493  0.014026  0.023081  ...  0.052281  0.031634
 2024-12-31  0.018035  0.044346  0.014094  0.023024  ...  0.052219  0.031564
 2025-01-02  0.018217  0.043978  0.014353  0.023129  ...  0.051910  0.031686
 2025-01-03  0.018103  0.044058  0.014287  0.023178  ...  0.051517  0.031761
```

For the Total portfolio, the same method is used as for each sub-portfolio, the sample results are following:

```python
optimal_daily_weights_all['Total']
```

✓ 0.0s

	WFC	ETN	AMZN	QCOM	LMT	KO	JNJ	ISRG	XOM	MDT	...	KKR
Date												
2024-01-02	0.009297	0.010140	0.009985	0.010761	0.004367	0.011188	0.005962	0.009317	0.004665	0.007553	...	0.014826
2024-01-03	0.009372	0.010118	0.009910	0.010494	0.004420	0.011424	0.006120	0.009194	0.004804	0.007639	...	0.014636
2024-01-04	0.009342	0.009972	0.009913	0.010402	0.004494	0.011566	0.006221	0.009039	0.004894	0.007750	...	0.014526
2024-01-05	0.009466	0.010040	0.009661	0.010303	0.004486	0.011538	0.006213	0.009079	0.004855	0.007825	...	0.014551
2024-01-08	0.009577	0.010044	0.009694	0.010332	0.004467	0.011506	0.006225	0.009046	0.004864	0.007875	...	0.014772
...	...	...	...	...	...	...	...	...	...	...	...	...
2024-12-27	0.010872	0.011330	0.011659	0.009399	0.003762	0.009494	0.004428	0.011629	0.003983	0.005994	...	0.021462
2024-12-30	0.010872	0.011248	0.011595	0.009407	0.003788	0.009563	0.004453	0.011649	0.004019	0.005985	...	0.021281
2024-12-31	0.010894	0.011307	0.011606	0.009360	0.003790	0.009613	0.004453	0.011614	0.004039	0.005972	...	0.021325
2025-01-02	0.010890	0.011313	0.011530	0.009320	0.003818	0.009668	0.004502	0.011528	0.004117	0.006005	...	0.021300
2025-01-03	0.010902	0.011336	0.011594	0.009338	0.003796	0.009620	0.004491	0.011594	0.004114	0.006041	...	0.021523

254 rows × 99 columns

Then I use the same method of Systematic and Idiosyncratic Return Decomposition, Carino Multi-Period Return Attribution, Risk Attribution in part1

And calculate Estimate the portfolio's **expected idiosyncratic risk** using the **residual variances provided by the CAPM** by using this formula:

$$\sigma_{\text{idio}}^{expected} = \sqrt{\sum_i w_i^2 \cdot \sigma_{\varepsilon_i}^2}$$

Total portfolio also use the same method

The following are the summaries of each portfolio and total portfolio:

```
===== Portfolio A Attribution Summary =====
Total Return        : 0.224619
  Systematic Return : 0.214041
  Idiosyncratic Ret.: 0.010579
Total Risk (Daily) : 0.008244
  Systematic Risk (Daily)    : 0.007930
  Idiosyncratic Risk (Daily): 0.000314
  Expected Idio Risk (Daily): 0.002619

===== Portfolio B Attribution Summary =====
Total Return        : 0.230978
  Systematic Return : 0.197342
  Idiosyncratic Ret.: 0.033636
Total Risk (Daily) : 0.007042
  Systematic Risk (Daily)    : 0.006866
  Idiosyncratic Risk (Daily): 0.000176
  Expected Idio Risk (Daily): 0.002143

===== Portfolio C Attribution Summary =====
Total Return        : 0.320950
  Systematic Return : 0.219626
  Idiosyncratic Ret.: 0.101323
Total Risk (Daily) : 0.008496
  Systematic Risk (Daily)    : 0.007859
  Idiosyncratic Risk (Daily): 0.000637
  Expected Idio Risk (Daily): 0.002298

===== Portfolio Total Attribution Summary =====
Total Return        : 0.261093
  Systematic Return : 0.209468
  Idiosyncratic Ret.: 0.051625
Total Risk (Daily) : 0.007577
  Systematic Risk (Daily)    : 0.007791
  Idiosyncratic Risk (Daily): -0.000214
  Expected Idio Risk (Daily): 0.001344
```

```
===== Return Attribution Summary =====
  Portfolio  Total Return  Systematic Return  Idiosyncratic Return
0         A        0.2246             0.2140                0.0106
1         B        0.2310             0.1973                0.0336
2         C        0.3209             0.2196                0.1013
3     Total        0.2611             0.2095                0.0516

===== Risk Attribution Summary =====
  Portfolio  Total Risk  Systematic Risk  Idiosyncratic Risk  \
0         A      0.0082           0.0079              0.0003
1         B      0.0070           0.0069              0.0002
2         C      0.0085           0.0079              0.0006
3     Total      0.0076           0.0078             -0.0002

   Expected Idio Risk
0              0.0026
1              0.0021
2              0.0023
3              0.0013
```

The attribution results demonstrate that:

Portfolio A achieved a total return of 22.46%, with 21.40% contributed by systematic return and a modest 1.06% from idiosyncratic sources. Portfolio B delivered a total return of 23.10%, with 19.73% from systematic return and 3.36% attributed to idiosyncratic performance. Portfolio C had the highest overall return of 32.10%, consisting of 21.96% systematic return and a strong 10.13% from idiosyncratic contributions.

Return Attribution Comparison

Total Returns: Part 2 shows significantly higher returns across all portfolios compared to Part1 Portfolio C demonstrates the strongest performance in both analyses (28.12% in Part 1 vs. 32.09% in Part 2),The magnitude difference between results ranges from 5.82-8.80 percentage points

Systematic vs. Idiosyncratic Return: Both results show systematic (market) returns as the primary return driver. Part 1 shows negative idiosyncratic return for Portfolio A (-5.24%), while Part 2 shows positive (1.06%)**.** Part 2 shows consistently higher idiosyncratic returns across all portfolios**.** Portfolio C maintains the highest idiosyncratic return in both analyses

Risk Attribution Comparison:

Total Risk Levels: Part 2 shows marginally higher total risk across portfolios A, B, and C

The Total portfolio risk remains virtually identical (0.0071 vs. 0.0076)

Risk Components: Both results identify systematic risk as the dominant factor (>90% of total risk)**.** Both show negative idiosyncratic risk for the Total portfolio, indicating diversification benefits **.** Part 2 shows lower idiosyncratic risk contributions for portfolios A and B, but slightly higher for portfolio C

Comparison Between Model-Predicted and Realized Idiosyncratic Risk:

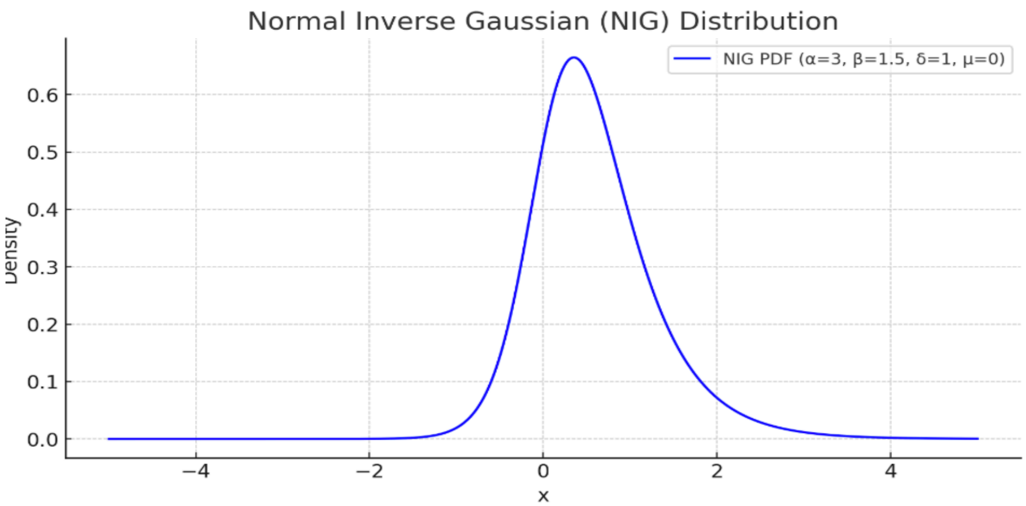
The CAPM model provided ex-ante expectations of idiosyncratic risk that differed from realized values: Expected idiosyncratic risks were: Portfolio A: 0.0026, Portfolio B: 0.0021,Portfolio C: 0.0023, Total portfolio: 0.0013. When comparing these expectations to realized values :All portfolios demonstrated lower realized idiosyncratic risk than expected, Portfolio C showed the smallest gap between expected (0.0023) and realized (0.0006), The Total portfolio showed an interesting result with negative realized idiosyncratic risk (-0.0002) compared to a positive expectation (0.0013) This suggests the CAPM model tends to overestimate idiosyncratic risk. The negative idiosyncratic risk for the Total portfolio indicates diversification benefits that reduce non-systematic risk beyond model predictions.

Question3:

Normal Inverse Gaussian:

The Normal Inverse Gaussian (NIG) distribution is a flexible model that captures both skewness and heavy tails, making it more suitable than the normal distribution for modeling financial returns.

Introduced by Barndorff-Nielsen in 1997, it is a special case of the Generalized Hyperbolic family and is often used in finance to describe the non-normal behavior of asset returns. Because it can reflect extreme events and asymmetry, The Normal Inverse Gaussian (NIG) distribution is particularly useful in quantitative risk management due to its ability to capture key stylized facts of financial return series, such as asymmetry (skewness) and heavy tails (kurtosis)—features that the normal distribution fails to represent. Modeling Jump Processes: The NIG distribution belongs to the generalized hyperbolic family, which allows it to model returns with sudden large jumps, a common feature in financial markets. Its infinite divisibility makes it suitable for Lévy process modeling, which is used in stochastic processes with jumps. Value at Risk (VaR) Calculation: Because NIG accommodates fat tails, it gives a more realistic estimation of tail risk than the normal distribution. This is crucial for accurate VaR and Expected Shortfall (ES) estimation, especially at high confidence levels where underestimation of risk can lead to severe consequences. Derivative Pricing: In incomplete markets or where returns are non-Gaussian, the NIG distribution provides a better fit to the observed market prices of options. It is particularly effective in modeling implied volatility smiles and skews, which are common in equity and FX markets.



From the chart we can see the Clear skewness (right-skewed, with the peak shifted to the left) Sharp peak and heavy tails, making it more capable than the normal distribution of capturing extreme movements in financial markets.

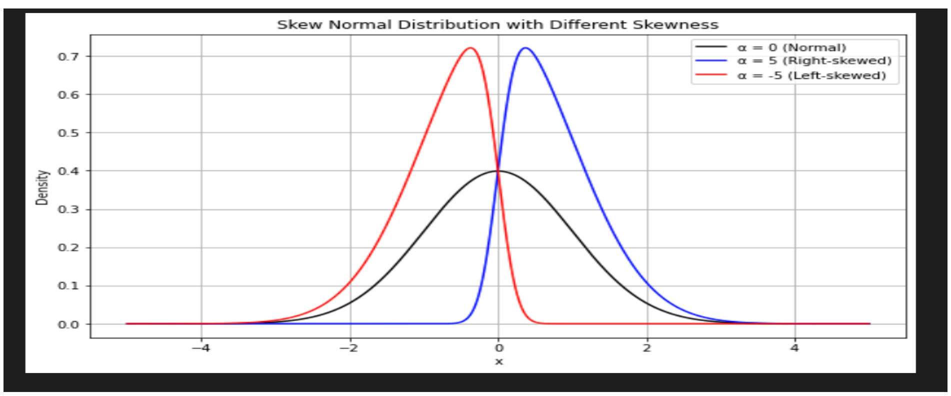
In Value at Risk (VaR) estimation, the traditional approach often assumes that asset returns follow a normal distribution. However, financial returns in reality frequently exhibit skewness and heavy tails—features that the normal distribution fails to capture. As a result, normal-based VaR calculations tend to underestimate the probability of extreme losses, leading to an overly optimistic assessment of risk.

In contrast, the Normal Inverse Gaussian (NIG) distribution accounts for both skewness and fat tails, making it better suited for modeling the actual behavior of asset returns. When using NIG to calculate VaR, the model assigns higher probability to tail events, which produces a more conservative and realistic risk estimate. This is particularly important in risk management, where underestimating the chance of large losses can have serious consequences.Therefore, replacing the normal assumption with NIG in VaR models enhances the robustness of risk assessments and helps institutions better prepare for extreme market events.

Skew Normal distributions:

The **Skew Normal distribution** is a generalization of the normal distribution that introduces a shape parameter to account for asymmetry. This allows it to model real-world financial return distributions that often exhibit skewness. Unlike the standard normal distribution, which assumes symmetry, the skew normal distribution can capture **longer left or right tails**, making it more flexible for risk modeling.

The sample application in finance: Financial Asset Return Modeling: Since actual asset returns often exhibit skewness (e.g., "Black swan" events in stock markets that lead to more frequent extreme negative returns), the skew normal distribution can more accurately describe the true return distribution, thereby improving modeling precision. Value at Risk (VaR) and Expected Shortfall (ES) Estimation: Compared to symmetric distributions, the skew normal distribution can better capture tail risks. At high confidence levels, it provides more conservative and realistic risk assessments, helping financial institutions develop robust risk control strategies. Option Pricing and Volatility Modeling: In the options market, implied volatility often exhibits a "smile" or "skew" pattern, reflecting the skewness in the underlying asset's return distribution. Modeling the underlying with a skew normal distribution can better explain the structure of implied volatility and improve the fit of pricing models.



The figure illustrates the probability density functions of the Skew Normal distribution under different skewness parameters (α). When $\alpha = 0$, the distribution is symmetric and reduces to the standard normal distribution, which assumes equal probabilities for extreme gains and losses. When $\alpha > 0$ (blue curve), the distribution becomes right skewed, with a longer right tail and the peak shifted to the left, indicating a higher chance of large positive returns. Conversely, when $\alpha < 0$ (red curve), the distribution is left-skewed, with a heavier left tail and the peak shifted to the right, representing a higher probability of extreme losses.

The flexibility of the **Skew Normal distribution** makes it particularly valuable in **quantitative risk control**, as financial asset returns in reality often exhibit **asymmetry**. Unlike the standard normal distribution, which assumes symmetric returns, the Skew Normal model introduces a skewness parameter (α) that allows it to represent **left-skewed** or **right-skewed** return profiles. This is especially important in markets where downside risks are more severe than upside gains.

When $\alpha < 0$, the model better reflects the reality where downside risk is greater than upside potential; When $\alpha > 0$, it is suitable for capturing situations where assets have explosive upside potential; When $\alpha = 0$, the distribution is identical to the normal distribution, returning to traditional risk modeling assumptions.

By adjusting α, the model can better reflect the **true shape of return distributions**, especially when modeling **left-tail risk**—the likelihood of extreme losses. This leads to **more accurate Value at Risk (VaR) estimates**, particularly in stress conditions where traditional models may underestimate risk. As a result, the Skew Normal distribution provides a **more realistic and robust framework** for assessing financial risk, making it a powerful alternative to the normal distribution in modern risk management practices.

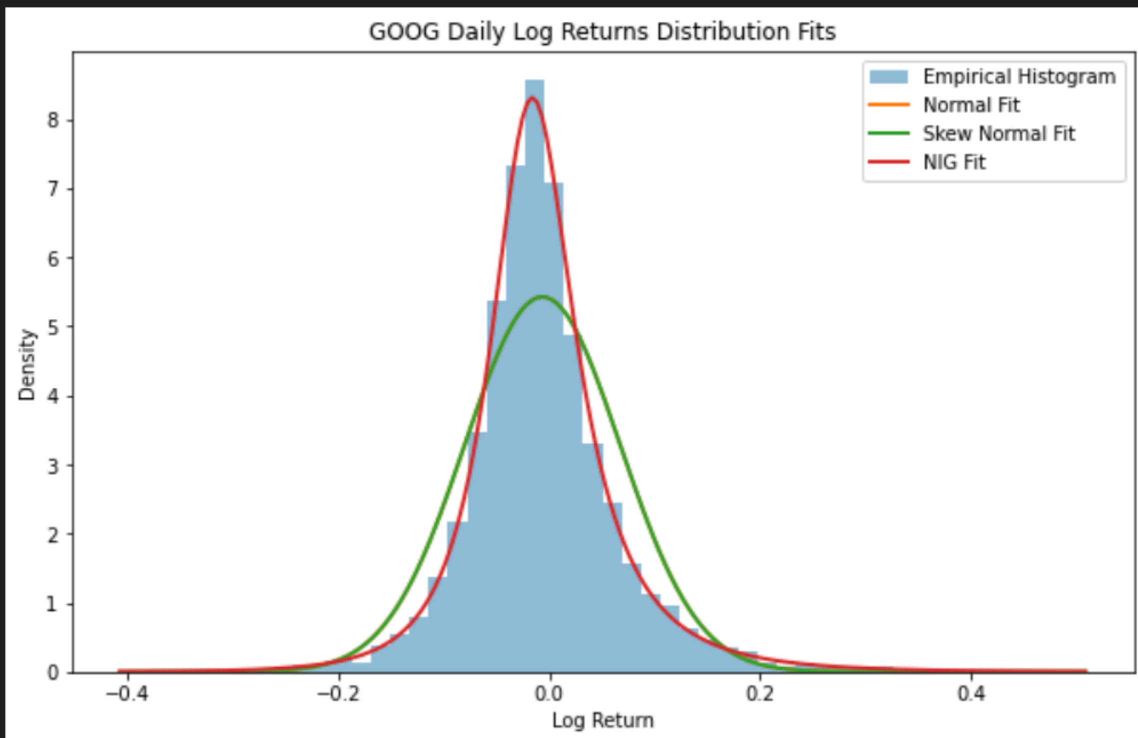
In practice, however, we find that the fitted skewness parameter α for UVXY daily log returns is approximately zero. This implies that the data is statistically symmetric and the Skew Normal distribution reduces to the standard Normal distribution. As a result, both distributions exhibit identical log-likelihoods, but Skew Normal incurs a higher AIC due to increased model complexity. You can see the example behind later.

Here is the example of UVXY: data from 2011-10-04 - 2025-04-14

```
=== Normal Distribution ===
mu = -0.006427, sigma = 0.073551
Log-likelihood = 4050.06
AIC = -8096.12

=== Skew Normal Distribution ===
alpha = 0.000000, loc = -0.006427, scale = 0.073551
Log-likelihood = 4050.06
AIC = -8094.12

=== NIG Distribution ===
alpha = 10.401597, beta = 2.385327, delta = 0.052854, mu = -0.018879
Log-likelihood = 4414.17
AIC = -8820.35
```



Choosing between them depends on whether the dominant risk feature is asymmetry (Skew Normal) or fat tails and jumps (NIG). When skewness is not statistically significant (e.g., $\alpha \approx 0$), the simpler Normal distribution may actually outperform Skew Normal in terms of parsimony. In our analysis of UVXY, the NIG distribution shows a clear advantage in capturing heavy tails, while Skew Normal collapses into Normal due to negligible skewness

In conclusion, while both Skew Normal and NIG offer improvements over the Normal distribution in theory, real-world application requires careful testing. Our results suggest that for UVXY, heavy tails are the dominant feature (captured by NIG), whereas skewness is not statistically significant.

Question4:

First, we need to calculate the returns for each stock, then extract the “estimation_data” from the dataset by using this formula:

$$R_t^i = \frac{P_t^i - P_{t-1}^i}{P_{t-1}^i}$$

```
estimation_returns
✓ 0.0s
```

Date	SPY	AAPL	NVDA	MSFT	AMZN	META	GOOGL	AVGO
2023-01-04	0.007720	0.010314	0.030318	-0.043743	-0.007924	0.021084	-0.011670	0.012214
2023-01-05	-0.011413	-0.010605	-0.032816	-0.029638	-0.023726	-0.003376	-0.021344	-0.009318
2023-01-06	0.022932	0.036794	0.041640	0.011785	0.035611	0.024263	0.013225	0.060196
2023-01-09	-0.000567	0.004089	0.051753	0.009736	0.014870	-0.004230	0.007786	-0.019612
2023-01-10	0.007013	0.004456	0.017981	0.007617	0.028732	0.027188	0.004544	-0.003398
...	...	...	...	...	...	...	...	...
2023-12-22	0.002010	-0.005547	-0.003266	0.002784	-0.002730	-0.001977	0.007620	-0.004710
2023-12-26	0.004223	-0.002841	0.009195	0.000214	-0.000065	0.004075	0.000212	0.008833
2023-12-27	0.001808	0.000518	0.002800	-0.001575	-0.000456	0.008455	-0.008126	-0.005054

Next, we fit the zero-mean returns of each stock to Normal, Student's t, Skew-Normal, and NIG distributions, by using following package : **scipy.stats.norm, scipy.stats.t, scipy.stats.skewnorm, scipy.stats.norminvgauss.** and for each stock’s historical returns, fit a “best” distribution and enforce a mean (loc) of zero in the process.

The historical return data in this project is relatively complete, with no significant missing values. However, when fitting return distributions for each stock, there is still a considerable difference in model complexity. For example, the normal distribution has only 2 parameters, while the NIG distribution can have up to 4. To more reasonably penalize the complexity of such models during the fitting process, we choose to use AICc. The formula is following:

$$\mathrm{AIC} = 2k - 2\ell, \quad \mathrm{AICc} = \mathrm{AIC} + \frac{2k^2 + 2k}{n - k - 1}$$

At the same time, By using inspect.signature(dist_obj._parse_args), we check whether each distribution includes a loc parameter. If it does, we set the corresponding value in the fitted parameter list to 0. This ensures that all fitted distributions have a mean of zero

The following are the sample results:

```
WFC: Best fit = Gen T (AICc=-1320.37), Params=[5.0037 0.      0.0137]
ETN: Best fit = Gen T (AICc=-1353.58), Params=[3.8783 0.      0.012 ]
AMZN: Best fit = Gen T (AICc=-1232.15), Params=[5.9219 0.      0.0169]
QCOM: Best fit = Gen T (AICc=-1259.42), Params=[5.2207 0.      0.0156]
LMT: Best fit = Gen T (AICc=-1589.31), Params=[3.7033 0.      0.0074]
KO: Best fit = Gen T (AICc=-1690.34), Params=[5.2155 0.      0.0066]
JNJ: Best fit = Gen T (AICc=-1611.97), Params=[3.605 0.     0.007]
ISRG: Best fit = Gen T (AICc=-1311.49), Params=[4.7002 0.      0.0137]
XOM: Best fit = Gen T (AICc=-1363.94), Params=[7.8807 0.      0.0136]
MDT: Best fit = Gen T (AICc=-1448.71), Params=[4.583  0.      0.0104]
DHR: Best fit = Gen T (AICc=-1395.12), Params=[5.3055 0.      0.0119]
PLD: Best fit = Gen T (AICc=-1337.94), Params=[6.6757 0.      0.0139]
BA: Best fit = Gen T (AICc=-1335.71), Params=[4.703  0.      0.0131]
PG: Best fit = Gen T (AICc=-1625.03), Params=[5.5198 0.      0.0076]
MRK: Best fit = Gen T (AICc=-1501.28), Params=[8.0684 0.      0.0103]
AMD: Best fit = Gen T (AICc=-1062.75), Params=[4.6975 0.      0.0226]
BX: Best fit = Gen T (AICc=-1200.40), Params=[6.3059 0.      0.0182]
PM: Best fit = Gen T (AICc=-1570.04), Params=[8.1562 0.      0.009 ]
SCHW: Best fit = Gen T (AICc=-1164.07), Params=[2.8391 0.      0.0159]
VZ: Best fit = Gen T (AICc=-1465.94), Params=[3.2712 0.      0.0091]
COP: Best fit = Gen T (AICc=-1307.26), Params=[5.8301 0.      0.0145]
ADI: Best fit = Gen T (AICc=-1348.73), Params=[6.3639 0.      0.0135]
BAC: Best fit = Gen T (AICc=-1338.24), Params=[4.2675 0.      0.0127]
NOW: Best fit = NIG (AICc=-1259.27), Params=[ 0.969  -0.2251  0.       0.0191]
TMO: Best fit = Gen T (AICc=-1415.77), Params=[5.1609 0.      0.0114]
CVX: Best fit = Gen T (AICc=-1418.55), Params=[4.5534 0.      0.011 ]
ANET: Best fit = Gen T (AICc=-1166.37), Params=[2.7441 0.      0.0156]
NVDA: Best fit = Gen T (AICc=-1086.71), Params=[4.7894 0.      0.0217]
GE: Best fit = NIG (AICc=-1381.76), Params=[6.1882 2.3721 0.      0.0334]
GILD: Best fit = Gen T (AICc=-1464.27), Params=[8.5693 0.      0.0112]
MU: Best fit = NIG (AICc=-1194.58), Params=[1.453  0.5481 0.      0.0248]
CMCSA: Best fit = Gen T (AICc=-1436.12), Params=[4.5561 0.      0.0106]
```

The next step is to precompute the quantile function, For each stock's marginal distribution, we precompute the quantile function (PPF) values in advance, so that during the Copula simulation we can use fast interpolation instead of calling `dist_obj.ppf(u)` point by point.
Each stock stores two attributes: `'u_values': u_values` and `'ppf_values': ppf_values`. This optimization is necessary because without it, the simulation could take over 10 minutes to complete. If there is inf or NaN, replace it with the median.

```
ppf_cache
1.6s
{'WFC': {'u_values': array([0.001     , 0.001999  , 0.002998  , 0.003997  , 0.004996  ,
       0.00599499, 0.00699399, 0.00799299, 0.00899199, 0.00999099,
       0.01098999, 0.01198899, 0.01298799, 0.01398699, 0.01498599,
       0.01598498, 0.01698398, 0.01798298, 0.01898198, 0.01998098,
       0.02097998, 0.02197898, 0.02297798, 0.02397698, 0.02497598,
       0.02597497, 0.02697397, 0.02797297, 0.02897197, 0.02997097,
       0.03096997, 0.03196897, 0.03296797, 0.03396697, 0.03496597,
       0.03596496, 0.03696396, 0.03796296, 0.03896196, 0.03996096,
       0.04095996, 0.04195896, 0.04295796, 0.04395696, 0.04495596,
       0.04595495, 0.04695395, 0.04795295, 0.04895195, 0.04995095,
       0.05094995, 0.05194895, 0.05294795, 0.05394695, 0.05494595,
       0.05594494, 0.05694394, 0.05794294, 0.05894194, 0.05994094,
       0.06093994, 0.06193894, 0.06293794, 0.06393694, 0.06493594,
       0.06593493, 0.06693393, 0.06793293, 0.06893193, 0.06993093,
       0.07092993, 0.07192893, 0.07292793, 0.07392693, 0.07492593,
       0.07592492, 0.07692392, 0.07792292, 0.07892192, 0.07992092,
       0.08091992, 0.08191892, 0.08291792, 0.08391692, 0.08491592,
       0.08591491, 0.08691391, 0.08791291, 0.08891191, 0.08991091,
       0.09090991, 0.09190891, 0.09290791, 0.09390691, 0.09490591,
       0.0959049 , 0.0969039 , 0.0979029 , 0.0989019 , 0.0999009 ,
       0.1008999 , 0.1018989 , 0.1028979 , 0.1038969 , 0.1048959 ,
       0.10589489, 0.10689389, 0.10789289, 0.10889189, 0.10989089,
       0.11088989, 0.11188889, 0.11288789, 0.11388689, 0.11488589,
       0.11588488, 0.11688388, 0.11788288, 0.11888188, 0.11988088,
       0.12087988, 0.12187888, 0.12287788, 0.12387688, 0.12487588,
       0.12587487, 0.12687387, 0.12787287, 0.12887187, 0.12987087,
       0.13086987, 0.13186887, 0.13286787, 0.13386687, 0.13486587,
       0.13586486, 0.13686386, 0.13786286, 0.13886186, 0.13986086,
       0.14085986, 0.14185886, 0.14285786, 0.14385686, 0.14485586,
       0.14585485, 0.14685385, 0.14785285, 0.14885185, 0.14985085,
```

This is the sample of precompute quantile function.

Next, we Based on the information in initial_portfolio.csv and the prices on December 31, 2023, we calculate the market value and weight of each stock within each portfolio. The market value of each stock is computed as: (number of shares held) × (stock price). Summing these gives the total portfolio value tot_val and dividing each stock's market value by tot_val yields its weight in the portfolio.

The next step is to calculate VaR/ES using the Copula model.

We start by extracting the subset of historical returns corresponding to each portfolio and compute the Spearman rank correlation matrix. By using following formula:

$$\rho_{\text{spearman}} = \operatorname{corr}(\operatorname{rank}(X), \operatorname{rank}(Y))$$

If the smallest eigenvalue is less than or equal to $1e{-}12$, it indicates that the matrix is nearly singular. In this case, a small value is added to the diagonal to ensure the matrix is positive definite.

$$\text{if } \lambda_{\min} \leq 1e - 12: \quad \rho_{\text{spearman}} \leftarrow \rho_{\text{spearman}} + \epsilon I$$

Next, we perform Cholesky decomposition on the Spearman rank correlation matrix for each portfolio. By uisng following formula:

$$\rho_{\text{spearman}} = L\,L^{\top}$$

At the same time, we generate Independent Standard Normal Samples. Construct a matrix Zraw~N(0,I) with shape (NSIM×d, where each element is drawn independently from a standard normal distribution. By using following formula:

$$Z_{\text{raw}} \sim N(0, I)$$

This ensures that the rows of the simulated matrix exhibit the same correlation structure as ρ spearman .

Then transform the independent samples into correlated variables by using following formula:

$$Z_{\text{corr}} = Z_{\text{raw}} \times L^{\top}$$

After that, we should conduct Uniform Distribution Convertion, Apply the CDF of the standard normal distribution Φ element-wise to obtain values uniformly distributed over [0,1] by using following formula:

$$U = \Phi(Z_{\text{corr}})$$

Then we should map to Marginal Distributions, for each stock j, transform the j-th column of uniform samples U j into the fitted marginal distribution using either the precomputed quantile function (PPF) or directly via dist_obj.ppf () by using following formula:

$$X_{\text{sim}}^{(j)} = F_j^{-1}(U^{(j)})$$

This produces joint return simulations that respect both the marginal distribution of each asset and the empirical dependence structure.

The portfolio return in each simulated time is calculated as the weighted sum of individual stock returns by uisng following formula:

$$R_{\text{port}} = X_{\text{sim}} \times w$$

Then we should calculate VAR and ES:

The first step is Sort the simulated portfolio returns in ascending order:

$$r_{(1)},\ r_{(2)},\ \cdots,\ r_{(N_{\text{SIM}})}$$

At confidence level α =0.05 the Value-at-Risk is defined as the negative of the α-quantile:

$$\text{VaR}_\alpha = -r_{(\lfloor \alpha N_{\text{SIM}} \rfloor)}$$

The ES is the average of all losses more extreme than VaR:

$$\text{ES}_\alpha = -\frac{1}{\lfloor \alpha N_{\text{SIM}} \rfloor + 1} \sum_{k=1}^{\lfloor \alpha N_{\text{SIM}} \rfloor + 1} r_{(k)}$$

Then we should conversion to Dollar:
If the portfolio value is V:

$$\text{Dollar VaR} = \text{VaR}_\alpha \times V, \quad \text{Dollar ES} = \text{ES}_\alpha \times V$$

We finally get the result summay:

	Portfolio	Method	Portfolio Value	VaR	ES	VaR(%)
0	A	Copula	295444.608200	4214.126898	5601.328470	1.426368
1	B	Copula	280904.482409	3685.306007	4898.606073	1.311943
2	C	Copula	267591.439955	3780.627132	4975.982111	1.412836

The Multivariate Normal Parametric Approach assumes that asset returns follow a joint multivariate normal distribution. I begin by computing the **covariance matrix** Sigma Σ of **mean-adjusted historical returns** by using following formula:

$$\Sigma = \text{Cov}(\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_T)$$

Next, i use the current portfolio holdings and prices to compute the total portfolio value V p and asset weights wj by using following formula:

$$V_p = \sum_{j=1}^{d} q_j \cdot P_j \quad , \quad w_j = \frac{q_j \cdot P_j}{V_p}$$

Then, we compute the portfolio standard deviation (volatility) using the portfolio weights and the submatrix of the covariance matrix corresponding to the selected assets by using following formula:

$$\sigma_p = \sqrt{\mathbf{w}^T \Sigma \mathbf{w}}$$

Next, we compute the **quantile** and **density** of the standard normal distribution at the confidence level α. Specifically:
The **quantile** zα formula:

$$z_\alpha = \Phi^{-1}(\alpha)$$

The **density** ϕ(zα) formula:

$$\phi(z_\alpha) = \frac{1}{\sqrt{2\pi}} e^{-z_\alpha^2/2}$$

Then I compute the Value at Risk and Expected Shortfall (ES) under the assumption that the portfolio return R p∼N(0,σ ^2p) The formulas are:

VaR (proportional):

$$\mathrm{VaR}_\alpha = -z_\alpha \cdot \sigma_p$$

ES (proportional):

$$\mathrm{ES}_\alpha = \frac{\phi(z_\alpha)}{\alpha} \cdot \sigma_p$$

To convert these to dollar-denominated risk values, we multiply by the current portfolio value Vp:

$$\mathrm{VaR}_\$ = \mathrm{VaR}_\alpha \cdot V_p = -z_\alpha \cdot \sigma_p \cdot V_p$$

$$\mathrm{ES}_\$ = \mathrm{ES}_\alpha \cdot V_p = \frac{\phi(z_\alpha)}{\alpha} \cdot \sigma_p \cdot V_p$$

The sample result is following:

3	A	MVN	295444.608200	4197.966533	5264.419394	1.420898
4	B	MVN	280904.482409	3668.822444	4600.851358	1.306075
5	C	MVN	267591.439955	3684.828997	4620.924221	1.377035

The summary of two method is following:

	Portfolio	Method	Portfolio Value	VaR	ES	VaR(%)	ES(%)
0	A	Copula	295444.608200	4214.126898	5601.328470	1.426368	1.895898
1	B	Copula	280904.482409	3685.306007	4898.606073	1.311943	1.743869
2	C	Copula	267591.439955	3780.627132	4975.982111	1.412836	1.859545
3	A	MVN	295444.608200	4197.966533	5264.419394	1.420898	1.781863
4	B	MVN	280904.482409	3668.822444	4600.851358	1.306075	1.637870
5	C	MVN	267591.439955	3684.828997	4620.924221	1.377035	1.726858

From the result, we observe that the Gaussian Copula approach now yields slightly higher VaR and ES estimates than the Multivariate Normal (MVN) method in some portfolios. In specific,
Portfolio A:
95% VaR under Copula is $4,214.13 (1.43%), compared to $4,197.97 (1.42%) under MVN.
ES under Copula is $5,601.33 (1.90%), which is higher than $5,264.42 (1.78%) from MVN.
For Portfolio B and C, the Copula approach also produces marginally higher tail risk estimates:
Portfolio B:
VaR: $3,685.31 (1.31%) vs. $3,668.82 (1.31%) , ES: $4,898.61 (1.74%) vs. $4,600.85 (1.64%)
Portfolio C: VaR: $3,780.63 (1.41%) vs. $3,684.83 (1.38%), ES: $4,975.98 (1.86%) vs. $4,620.92 (1.73%)

The Copula approach leverages the empirical marginal distributions of each stock, maintaining characteristics like skewness, fat tails, and non-linear features. It models dependence using Spearman rank correlation and simulates joint outcomes through a Gaussian Copula, making it highly suitable for capturing realistic co-movements under market stress.

The MVN approach, on the other hand, assumes a joint multivariate normal distribution for asset returns. While computationally convenient, this method is limited by the normality assumption and often underestimates or overestimates tail risk depending on data structure. In this case, the fact that Copula VaR/ES is now higher reflects: A more accurate representation of non-normal behavior in marginal distributions; Better modeling of extreme co-movement and tail dependence across assets.

To sum up, the result underscore the flexibility and robustness of the **Gaussian Copula method** in risk assessment. By incorporating both **empirical distributional features** and **rank-based dependency structures**, it produces **risk measures that more accurately reflect actual market behavior**, particularly in the tails. While MVN remains a useful benchmark, it may fall short in capturing the complex joint dynamics of financial returns.

Then I use the same method of copula and MVN for Total portfolio, and the sample results are following:

```
Total Copula: VaR=1.3506% ($11398.15), ES=1.8092% ($15268.68)
Total MVN:    VaR=1.3254% ($11185.53), ES=-1.6621% ($-14027.11)
```

Total Portfolio risk is consistent with the behavior observed in the sub-portfolios. After aggregating all holdings, the 95 % confidence Copula VaR is 1.350 % (≈ $11,398) and ES is 1.809 % (≈ $15,269). The MVN method yields a VaR of 1.325 % (≈ $11,186) and an ES of 1.662 % (≈ $14,027). Once again, Copula produces slightly higher tail-risk estimates than MVN, indicating a more conservative stance when capturing cross-asset tail dependence. This reinforces that, at the whole-portfolio level, combining empirical marginal distributions with rank-based dependence via a Copula framework offers a more realistic view of potential losses under extreme market conditions.

Question 5:
From my previous work, we can directly obtain the following variables without needing to find the optimal distribution again:

df_portfolio: A dataframe containing portfolio information, showing which assets are included in each portfolio
fitting_records: Records of the optimal probability distribution parameters already fitted for each asset
estimation_returns: Dataset containing historical asset returns
corr_dict: A dictionary used to store the correlation matrix and corresponding asset list for each portfolio

First, for each sub-portfolio, we construct the Spearman correlation matrix based on estimation period returns. By using following formula:

$$\Sigma_{i,j} = \mathrm{Corr}_{\mathrm{Spearman}}(r_i, r_j)$$

Since Cholesky decomposition requires positive definiteness, we apply a minimum eigenvalue adjustment to ensure the matrix is valid. By using following formula:

$$\Sigma \leftarrow \Sigma + \delta I$$

$$\delta = \max(0, \epsilon - \lambda_{\min})$$

We can prepare Spearman correlation for later calculation.

Then we should do the Copula simulation and ES calculation , The steps of Copula simulation and ES calculation are the same as those of the previous questions

We use a Gaussian Copula to simulate the joint return distribution of assets. Independent standard normal samples are generated

$$Z_{\mathrm{corr}} = Z L^T, \quad \Sigma = L L^T$$

and then mapped to a joint uniform distribution.

$$U_{ij} = \Phi(Z_{\mathrm{corr},ij})$$

These are then transformed back into asset returns using the PPF interpolation of the fitted marginal distributions.

$$X_{ij} = F_i^{-1}(U_{ij})$$

Finally, we simulate the portfolio returns and compute both VaR and ES.

$$R_{p,t} = \sum_{i=1}^{n} w_i X_{ti}$$

$$\mathrm{VaR}_\alpha = \mathrm{Quantile}_\alpha(R_p), \quad \mathrm{ES}_\alpha = -\mathbb{E}[R_p \mid R_p \leq \mathrm{VaR}_\alpha]$$

The next step is to calculate risk contributions and perform risk parity optimization. For each asset *i*, its marginal risk contribution to the portfolio's ES is calculated by following formula:

$$RC_i = -w_i \cdot \mathbb{E}[r_i \mid R_p \leq \mathrm{VaR}_\alpha]$$

The objective is to equalize all RCi, the Objective function is following:

$$\mathcal{L}(w) = \sum_{i=1}^{n} \left(RC_i - \frac{\mathrm{ES}}{n} \right)^2$$

. The optimization is then subject to the following constraints:

$$\sum_{i=1}^{n} w_i = 1, \quad 0 \leq w_i \leq 1$$

The sample results are following:

```
===== Constructing Risk Parity Portfolios =====

Optimizing Risk Parity for Portfolio A
  Optimization success - Portfolio ES: 0.018773
  Risk Contributions:
    WFC: weight=0.0301, risk_contrib=0.000697, pct=3.71%
    ETN: weight=0.0300, risk_contrib=0.000650, pct=3.46%
    AMZN: weight=0.0319, risk_contrib=0.000795, pct=4.24%
    QCOM: weight=0.0311, risk_contrib=0.000965, pct=5.14%
    LMT: weight=0.0320, risk_contrib=0.000206, pct=1.10%
    KO: weight=0.0292, risk_contrib=0.000189, pct=1.00%
    JNJ: weight=0.0294, risk_contrib=0.000211, pct=1.13%
    ISRG: weight=0.0300, risk_contrib=0.000721, pct=3.84%
    XOM: weight=0.0350, risk_contrib=0.000471, pct=2.51%
    MDT: weight=0.0329, risk_contrib=0.000455, pct=2.43%
    DHR: weight=0.0283, risk_contrib=0.000481, pct=2.56%
    PLD: weight=0.0272, risk_contrib=0.000613, pct=3.27%
    BA: weight=0.0294, risk_contrib=0.000562, pct=2.99%
    PG: weight=0.0282, risk_contrib=0.000188, pct=1.00%
    MRK: weight=0.0309, risk_contrib=0.000128, pct=0.68%
    AMD: weight=0.0322, risk_contrib=0.001204, pct=6.41%
    BX: weight=0.0307, risk_contrib=0.000972, pct=5.18%
    PM: weight=0.0271, risk_contrib=0.000253, pct=1.35%
    SCHW: weight=0.0323, risk_contrib=0.000981, pct=5.23%
    VZ: weight=0.0334, risk_contrib=0.000300, pct=1.60%
    COP: weight=0.0282, risk_contrib=0.000444, pct=2.36%
    ADI: weight=0.0264, risk_contrib=0.000654, pct=3.48%
    BAC: weight=0.0314, risk_contrib=0.000807, pct=4.30%
    NOW: weight=0.0305, risk_contrib=0.001038, pct=5.53%
    TMO: weight=0.0310, risk_contrib=0.000543, pct=2.89%
    CVX: weight=0.0255, risk_contrib=0.000327, pct=1.74%
    ANET: weight=0.0279, risk_contrib=0.000763, pct=4.06%
    NVDA: weight=0.0338, risk_contrib=0.001160, pct=6.18%
    GE: weight=0.0342, risk_contrib=0.000077, pct=0.41%
    GILD: weight=0.0311, risk_contrib=0.000339, pct=1.81%
    MU: weight=0.0286, risk_contrib=0.000434, pct=2.31%
    CMCSA: weight=0.0246, risk_contrib=0.000410, pct=2.19%
```

Then we should do the holding Period Return Simulation (Buy-and-Hold) the steps is same to the previous question:

1 Initial Portfolio Weights

2 At each time ttt, the portfolio return is computed as the weighted sum of asset returns:

$$R_{p,t} = \sum_{i=1}^{n} w_{i,t-1} \cdot r_{i,t}$$

3 Portfolio weights are updated based on the relative change in asset value

$$w_{i,t} = \frac{w_{i,t-1}(1 + r_{i,t})}{\sum_{j=1}^{n} w_{j,t-1}(1 + r_{j,t})}$$

This process captures a passive, self-financing strategy where weights evolve naturally according to asset performance, without rebalancing.

The sample results are the followings:

```
risk_parity_port_returns
✓ 0.0s
{'A': Date
 2024-01-02   -0.001296
 2024-01-03   -0.008940
 2024-01-04    0.000191
 2024-01-05    0.004070
 2024-01-08    0.011124
                 ...
 2024-12-27   -0.008093
 2024-12-30   -0.011123
 2024-12-31   -0.001214
 2025-01-02    0.003562
 2025-01-03    0.015336
 Length: 254, dtype: float64,
 'B': Date
 2024-01-02   -0.001680
 2024-01-03   -0.005783
 2024-01-04   -0.000134
 2024-01-05    0.001149
 2024-01-08    0.011038
                 ...
 2024-12-27   -0.007755
 2024-12-30   -0.011679
 2024-12-31   -0.000540
 2025-01-02   -0.000615
 2025-01-03    0.006298
 Length: 254, dtype: float64,
 'C': Date
 2024-01-02   -0.006916
 2024-01-03   -0.009068
 2024-01-04   -0.000947
 2024-01-05    0.000077
 2024-01-08    0.012431
```

```
risk_parity_daily_weights
✓ 0.0s
{'A':            WFC       ETN      AMZN      QCOM       LMT       KO  \
 Date
 2024-01-02  0.030073  0.029961  0.031865  0.031125  0.031984  0.029225
 2024-01-03  0.030179  0.029760  0.031484  0.030217  0.032229  0.029705
 2024-01-04  0.030050  0.029300  0.031459  0.029918  0.032734  0.030043
 2024-01-05  0.030415  0.029468  0.030626  0.029602  0.032639  0.029937
 2024-01-08  0.030685  0.029396  0.030644  0.029603  0.032409  0.029771
 ...              ...       ...       ...       ...       ...       ...
 2024-12-27  0.036248  0.034507  0.038352  0.028021  0.028401  0.025562
 2024-12-30  0.036212  0.034223  0.038103  0.028019  0.028573  0.025722
 2024-12-31  0.036259  0.034375  0.038110  0.027855  0.028564  0.025836
 2025-01-02  0.036216  0.034364  0.037827  0.027716  0.028751  0.025963
 2025-01-03  0.036061  0.034252  0.037835  0.027621  0.028431  0.025697
```

Then I applied the same return attribution and risk attribution methods used in Question 1 to decompose the risk of the portfolio constructed in Question 5, and obtained the following results.

```
===== Risk Parity Portfolio Return Attribution Summary =====
  Portfolio            Method  Total Return  Systematic Return
0         A  Risk Parity (ES)        0.2394             0.2087
1         B  Risk Parity (ES)        0.2559             0.1860
2         C  Risk Parity (ES)        0.3972             0.2229
3     Total  Risk Parity (ES)        0.2975             0.2058

   Idiosyncratic Return
0                0.0306
1                0.0698
2                0.1744
3                0.0917

===== Risk Parity Portfolio Risk Attribution Summary =====
  Portfolio            Method  Total Risk  Systematic Risk  \
0         A  Risk Parity (ES)      0.0082           0.0077
1         B  Risk Parity (ES)      0.0068           0.0064
2         C  Risk Parity (ES)      0.0088           0.0078
3     Total  Risk Parity (ES)      0.0076           0.0077

   Idiosyncratic Risk
0              0.0005
1              0.0004
2              0.0010
3             -0.0001
```

I applied return and risk attribution to the Risk Parity (ES) portfolios constructed in Question 5. The results show that Portfolio C achieved the highest total return (39.72%), largely driven by its idiosyncratic return (17.44%), while Portfolio A had the lowest (23.94%). Systematic return contributions were relatively stable across portfolios, ranging from 18.60% to 22.29%. In terms of risk, all portfolios exhibited low total volatility (between 0.0068 and 0.0088), with most of the risk attributed to systematic sources. Notably, the Total portfolio's idiosyncratic risk was approximately zero, suggesting strong diversification effects across sub-portfolios.

Portfolio C shows the highest idiosyncratic return, along with the highest idiosyncratic risk, reflecting a clear risk–return tradeoff. In contrast, Portfolios A and B appear more conservative. Notably, the Total portfolio has near-zero idiosyncratic risk, highlighting strong diversification across sub-portfolios. While ES-based optimization reduces tail risk, systematic risk still dominates, indicating continued sensitivity to market movements.

In terms of total return, the Risk Parity (ES) portfolios in Part 5 outperform those in both Part 1 and Part 2. Portfolio C, in particular, achieved a return of **39.72%**, exceeding its Part 1 return (28.12%) and even the optimized return in Part 2 (32.09%). This suggests that tail-risk control via Expected Shortfall can, in practice, lead to superior realized performance—even without explicitly optimizing for expected return. Additionally, the Total portfolio in Part 5 delivered **29.75%**, outperforming both Part 1 (20.47%) and Part 2 (26.11%), indicating robust performance across sub-portfolios.

In Part 1, Portfolio A had a **negative idiosyncratic return (–5.24%)**, and Portfolios B and C showed only marginal idiosyncratic gains. This implies that the original portfolios were dragged down by idiosyncratic components. Part 2 improved total returns but largely through increased systematic exposure, with limited idiosyncratic contribution. In contrast, all portfolios in Part 5 exhibit positive idiosyncratic returns, with Portfolio C contributing as much as **17.44%**. This indicates that the ES-based construction effectively captures and leverages non-market sources of excess return.

From a risk attribution perspective, the Total portfolio in Part 5 exhibits an overall volatility of **0.0076**, with **near-zero idiosyncratic risk (–0.0001)**. In contrast, Part 1 and Part 2 showed idiosyncratic risks of 0.0001 and 0.0013, respectively. This implies that the Part 5 construction achieved significantly stronger diversification across sub-portfolios, effectively neutralizing non-systematic noise. Moreover, systematic risk dominates across all portfolios in Part 5 (over 95%), similar to Part 1. However, unlike Part 1, Part 5 successfully converts idiosyncratic exposures into positive returns rather than performance drags.

In summary, the ES-based Risk Parity strategy in Part 5 delivers stronger idiosyncratic returns, lower idiosyncratic risk, and greater portfolio stability—while effectively mitigating tail risk.
It is particularly suitable for risk-sensitive, long-term allocation objectives.
Compared to the original portfolios in Part 1 and the Sharpe-maximizing portfolios in Part 2, Part 5 demonstrates superior **tail-adjusted return efficiency**.