Carrie_Little_AAI_500_FinalProject_Risk Parity

October 3, 2024

1 Risk Parity - Opportunity dataset

Carrie Little

1.0.1 Import Necessary Libraries

[1]:
```
# Carrie Little - AAI5000 Final Project Code
#
# Import All Necessary Libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import scipy.stats as stats
import cvxpy as cp
```

1.0.2 Models

[2]:
```
# Carrie Little - AAI5000 Final Project Code
#
# Risk Parity Optimization Model

# Load Opportunity Dataset
data = pd.read_csv('Opportunity_Set.csv')       # Load Dataset as Dataframe
data.head()
 ↪       # Diaplay 1st 5 in Dataframe
```

[2]:
```
         Date  Vanguard LifeStrategy Income Fund (VASIX)  \
0  11/30/2014                                     0.0094
1  12/31/2014                                    -0.0005
2   1/31/2015                                     0.0141
3   2/28/2015                                     0.0033
4   3/31/2015                                     0.0018

   Vanguard Total World Stock ETF (VT)  \
0                               0.0126
1                              -0.0199
2                              -0.0163
```

```
3                              0.0595
4                             -0.0121

   PIMCO 25+ Year Zero Coupon US Trs ETF (ZROZ)  \
0                                       0.0414
1                                       0.0612
2                                       0.1600
3                                      -0.1007
4                                       0.0117

   AQR Diversified Arbitrage I (ADAIX)  iShares Gold Trust (IAU)  \
0                              -0.0066                   -0.0053
1                              -0.0117                    0.0133
2                              -0.0059                    0.0865
3                               0.0069                   -0.0579
4                               0.0000                   -0.0222

   Bitcoin Market Price USD (^BTC)  \
0                           0.0969
1                          -0.1777
2                          -0.2677
3                           0.1062
4                          -0.0150

   AQR Risk-Balanced Commodities Strategy I (ARCIX)  \
0                                          -0.0726
1                                          -0.0412
2                                          -0.0287
3                                           0.0044
4                                          -0.0573

   AQR Long-Short Equity I (QLEIX)  AQR Style Premia Alternative I (QSPIX)  \
0                           0.0248                                  0.0412
1                           0.0140                                  0.0002
2                           0.0156                                 -0.0112
3                           0.0236                                 -0.0390
4                          -0.0027                                  0.0256

   AQR Equity Market Neutral I (QMNIX)  AQR Macro Opportunities I (QGMIX)  \
0                               0.0257                            0.0154
1                               0.0195                            0.0039
2                               0.0290                           -0.0070
3                              -0.0078                            0.0091
4                               0.0049                            0.0261

   AGF U.S. Market Neutral Anti-Beta (BTAL)  \
0                                    0.0235
```

```
1                                    0.0294
2                                    0.0320
3                                   -0.0568
4                                    0.0000

   AQR Managed Futures Strategy HV I (QMHIX)  \
0                                     0.1159
1                                     0.0461
2                                     0.0721
3                                    -0.0108
4                                     0.0655

   Invesco DB US Dollar Bullish (UUP)  ProShares VIX Mid-Term Futures (VIXM)
0                              0.0165                                -0.0298
1                              0.0213                                 0.0553
2                              0.0484                                 0.0762
3                              0.0028                                -0.1145
4                              0.0278                                 0.0033
```

[3]:

```
# Extract the returns data (excluding the Date column)
returns = data.iloc[:, 1:]

# Calculate the covariance matrix of asset returns
cov_matrix = returns.cov()

# Number of assets
n_assets = cov_matrix.shape[0]

# Define the variables for the optimization (portfolio weights)
weights = cp.Variable(n_assets)

# Define the objective (minimize portfolio variance)
portfolio_variance = cp.quad_form(weights, cov_matrix.values)

# Constraints (weights sum to 1 and are non-negative)
constraints = [cp.sum(weights) == 1, weights >= 0]

# Optimization problem (minimize variance)
problem = cp.Problem(cp.Minimize(portfolio_variance), constraints)
problem.solve()

# Optimal portfolio weights
optimal_weights = weights.value

# Compute Marginal Risk Contribution (MRC)
mrc = 2 * np.dot(cov_matrix.values, optimal_weights)
```

```python
# Creating a dataframe to display the results
mrc_df = pd.DataFrame({
    'Asset': returns.columns,
    'Optimal Weights': optimal_weights,
    'Marginal Risk Contribution': mrc
})

# Display the results
mrc_df
```

[3]:

```
                                                 Asset  Optimal Weights  \
0           Vanguard LifeStrategy Income Fund (VASIX)     3.932301e-01
1                 Vanguard Total World Stock ETF (VT)    -2.099375e-18
2      PIMCO 25+ Year Zero Coupon US Trs ETF (ZROZ)    -5.248749e-18
3                 AQR Diversified Arbitrage I (ADAIX)     1.133455e-01
4                            iShares Gold Trust (IAU)    -2.448059e-18
5                     Bitcoin Market Price USD (^BTC)     1.844191e-18
6   AQR Risk-Balanced Commodities Strategy I (ARCIX)     2.807962e-02
7                     AQR Long-Short Equity I (QLEIX)    -5.006550e-19
8            AQR Style Premia Alternative I (QSPIX)     4.818605e-19
9               AQR Equity Market Neutral I (QMNIX)     7.008853e-02
10                 AQR Macro Opportunities I (QGMIX)     1.092522e-01
11         AGF U.S. Market Neutral Anti-Beta (BTAL)     3.719750e-02
12         AQR Managed Futures Strategy HV I (QMHIX)    -9.423550e-19
13               Invesco DB US Dollar Bullish (UUP)     2.149796e-01
14             ProShares VIX Mid-Term Futures (VIXM)     3.382701e-02

    Marginal Risk Contribution
0                     0.000061
1                     0.000079
2                     0.000239
3                     0.000061
4                     0.000098
5                     0.000238
6                     0.000061
7                     0.000089
8                     0.000098
9                     0.000061
10                    0.000061
11                    0.000061
12                    0.000131
13                    0.000061
14                    0.000061
```

[4]:

```python
# Assuming equal weights for simplicity in this example, but you can replace
 ↪with any other weighting strategy
n_assets = len(returns.columns)
```

```
equal_weights = np.array([1/n_assets] * n_assets)

# Calculate the portfolio variance
portfolio_variance_equal_weight = equal_weights.T @ cov_matrix @ equal_weights

# Portfolio variance result
portfolio_variance_equal_weight
```

[4]: 0.0003504452730310492

[5]:
```
# Calculate the mean returns (expected returns) of each asset
expected_returns = returns.mean()

# Calculate the portfolio's expected return using equal weights
portfolio_expected_return_equal = np.dot(equal_weights, expected_returns)

# Portfolio expected return result
portfolio_expected_return_equal
```

[5]: 0.007898983050847456

[6]:
```
# Calculate the mean returns (expected returns) of each asset
expected_returns = returns.mean()

# Calculate the portfolio's expected return using optimal weights
portfolio_expected_return_optimal = np.dot(optimal_weights, expected_returns)

# Portfolio optimal return result
portfolio_expected_return_optimal
```

[6]: 0.0028506210338368025

[7]:
```
# Assume risk-free rate is 0 for simplicity, you can adjust as necessary
risk_free_rate = 0.0

# Portfolio variance and standard deviation
portfolio_variance_optimal = np.dot(optimal_weights.T, np.dot(cov_matrix,␣
 ↪optimal_weights))
portfolio_std = np.sqrt(portfolio_variance_optimal)

# Sharpe ratio calculation
portfolio_sharpe_ratio = (portfolio_expected_return_optimal - risk_free_rate) /␣
 ↪portfolio_std

# Display the Sharpe ratio
print(f"Portfolio Sharpe Ratio: {portfolio_sharpe_ratio:.4f}")
```

```
Portfolio Sharpe Ratio: 0.5141
```

[8]:
```
print("Equal Weighted Portfolio")
print(f"The Portfolio Variance is {portfolio_variance_equal_weight:.4f}")
print(f"The Expected Return is {portfolio_expected_return_equal:.4f}")
print()
print("Optimal Weighted Portfolio")
print(f"The Portfolio Variance is {portfolio_variance_optimal:.4f}")
print(f"The Expected Return is {portfolio_expected_return_optimal:.4f}")
print(f"The Portfolio Sharpe Ratio is {portfolio_sharpe_ratio:.4f}")
```

```
Equal Weighted Portfolio
The Portfolio Variance is 0.0004
The Expected Return is 0.0079

Optimal Weighted Portfolio
The Portfolio Variance is 0.0000
The Expected Return is 0.0029
The Portfolio Sharpe Ratio is 0.5141
```

1.0.3 Need to figure out how to remove negative

References

Agresti, Alan, and Maria Kateri. Foundations of Statistics for Data Scientists: With R and Python. CRC Press, Taylor & Francis Group, 2022.

Agresti, Alan, and Maria Kateri. (2022) Appendix B2. Chapter 2: Python for Probability Distributions. In Foundations of Statistics for Data Scientists: With R and Python (p. 385-389). CRC Press, Taylor & Francis Group, 2022.

ChatGPT, (2024) GPT-4o version, OpenAI. [Large language model]. https://chatgpt.com/

Opportiunty Dataset - need link/website info

Fama French Factors , Kenneth French's website. http://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_

[ ]: