I. Introduction

1)Topic and Purpose of Research

In the field of finance, the way a portfolio is constructed plays a crucial role in achieving optimal investment outcomes. Traditional methods, such as mean-variance optimization and risk-parity, have long been used by practitioners and financial institutions. On the other hand, methods involving machine learning methods such as clustering have gained significant attention in recent years as potential tools for portfolio construction. Regardless of which method is used, investors want their portfolio to be optimally diversified and in turn generate stable returns. Thus, the robustness of a portfolio is usually one of its most important evaluation metrics.

The purpose of our research is to investigate the in-sample and out-of-sample stability of both traditional and machine learning-powered portfolio allocation approaches using simulated data from the multivariate normal distribution with random drift and correlation matrices from the CorrGAN model (Marti, 2022), which is based on a deep learning model called Generative Adversarial Networks. This model can generate realistic correlation matrices which can exhibit multiple properties of the correlation matrices in the real financial data, one of which is the hierarchical structure of the correlations (shown below).

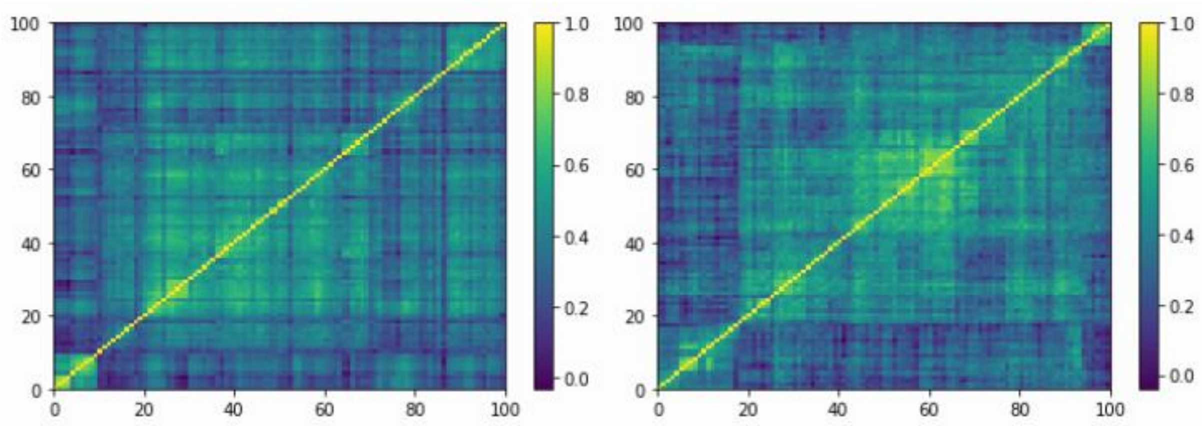

Figure 1 (Left) Empirical correlation matrix estimated on stock returns; (Right) GAN-generated correlation matrix (Marti 2020b).

The model was trained on approximately 10,000 empirical correlation matrices estimated on company returns included in S&P 500, which were sorted by a permutation induced by a hierarchical clustering algorithm. The below table shows the architecture of the generator part of the CorrGAN model.

Layer (type)	Output Shape	Param #
dense (Dense)	(None, 6400)	640000
batch_normalization (BatchNormalization)	(None, 6400)	25600
leaky_re_lu (LeakyReLU)	(None, 6400)	0
reshape (Reshape)	(None, 5, 5, 256)	0
conv2d_transpose (Conv2DTranspose)	(None, 10, 10, 128)	294912
batch_normalization_1 (BatchNormalization)	(None, 10, 10, 128)	512
leaky_re_lu_1 (LeakyReLU)	(None, 10, 10, 128)	0
conv2d_transpose_1 (Conv2DTranspose)	(None, 50, 50, 64)	73728
batch_normalization_2 (BatchNormalization)	(None, 50, 50, 64)	256
leaky_re_lu_2 (LeakyReLU)	(None, 50, 50, 64)	0
conv2d_transpose_2 (Conv2DTranspose)	(None, 50, 50, 1)	576

Total params: 1,035,584
Trainable params: 1,022,400
Non-trainable params: 13,184

Table 1: The Generator Architecture of the CorrGAN Model

We will use the already trained model, which is publicly available for different matrix dimensions (in our case, it is 50).With such a tool, we can test how portfolio allocation methods might perform in different but realistic scenarios.

It is clear that any conclusion will be dependent on the generated data, and might not fully generalize to real data, however we hope that Monte Carlo experiments that we will perform will yield to a better understanding of the numerical behavior of these portfolio allocation methods than applying and comparing them directly on arbitrarily selected real dataset.

Traditional portfolio allocation methods that we are going to look at are the most common ones: Equal Weight (EW), Minimum-Variance (MV), and Equal Risk Contribution (ERC) portfolio allocations methods. The EW allocation method completely ignores the correlation structure between assets, and distributes the weights equally across all assets, which are kept constant through time. The MV allocation considers the correlation structure between assets, and it seeks to find the combination of asset weights that result in the lowest portfolio variance, which involves the inversion of the covariance matrix, an operation that might be highly sensitive to estimation errors.

Unlike the MV, the ERC method seeks to achieve an equal risk allocation across all assets, even though it does not lead to minimum variance. The ERC method can be thought of as generally falling somewhere in between the EW portfolio on the one end, and the MV portfolio on the other (Maillard et al., 2010). Minimum variance allocation can result in some assets getting very little or even zero weight, while the ERC will only do so in very situations. Hence ERC can be more robust to economic shocks and have less idiosyncratic risk compared to MV.

Machine learning based portfolio construction approaches that we are going to focus on are Hierarchical Risk Parity (HRP) and Hierarchical Equal Risk Contribution (HERC). The Hierarchical Risk Parity is a method to perform asset allocation without the need to invert a covariance matrix (Lopez De Prado, 2016). Using a hierarchical clustering algorithm, it computes a hierarchical tree from the correlation matrix transformed using a distance metric, and based on that it groups similar investments into clusters. After the clustering step, it reorganizes the rows and columns of the covariance matrix, so that the largest values lie along the diagonal. Then, it performs top-down weight allocation using inverse-variance weighting method for each subtree in the hierarchical clustering tree. The method was invented to address the limitations of traditional risk parity methods by considering both the pairwise correlations and the hierarchical structure of assets. By incorporating the hierarchical relationships, the method allows for a more accurate and efficient risk allocation, potentially enhancing the risk diversification within the portfolio.

The Hierarchical Equal Risk Contribution (HERC) method is a portfolio allocation technique that combines the concepts of hierarchical structure and equal risk contribution (Raffinot, T., 2018). While the HRP method considers the hierarchical structure but does not necessarily aim for equal risk contributions, the ERC method focuses on equal risk contributions but does not explicitly incorporate hierarchical relationships. The HERC method took ideas from both the HRP and Equal Risk Contribution (ERC) methods, meaning that it uses the hierarchical structure in the correlation matrix and equal risk contribution principles to allocate weights to assets in a portfolio.

By examining the above-mentioned portfolio allocation approaches, we aim to find out which method performs better than others in terms of stability, which will be measured by the portfolio volatility. Both in-sample and out-of-sample results will be evaluated, with more weight given to the out-of-sample performance.

II.

Main Subject

1) Research Planning

We will first generate 1000 correlation matrices using CorrGAN, where each correlation matrix will be 50x50 (we decide to sample lower dimensional matrices because it would take much longer time to generate higher dimensional matrices such as 100x100 or 200x200). After generative sampling, we will preprocess those correlation matrices so that they become symmetric, and the diagonal terms are equal to 1. Then, we will generate 50 return series from the multivariate Gaussian distribution parametrized by zero means and a synthetically generated correlation matrix.

We will examine two cases: 1) return series without random shocks and 2) return series with random shocks. Random shocks, which will have the same dimension as a vector of sampled returns, will be generated independently for each return series from the student's t-distribution with zero mean, same standard deviation as that of corresponding return series and 4 degrees of freedom. The entries in the generated random shock vector will be added to the corresponding entries in the return series if the individual shock amount exceeds 4 standard deviations but is less than 6 standard deviations. This is done to reduce the number and scale of shocks in the return series.

The graphs below show an example of 50 return series generated from the multivariate Gaussian distribution **(without and with shocks, respectively)** with zero means and parametrized by a synthetic correlation matrix from CorrGAN for one trading year (252 days).

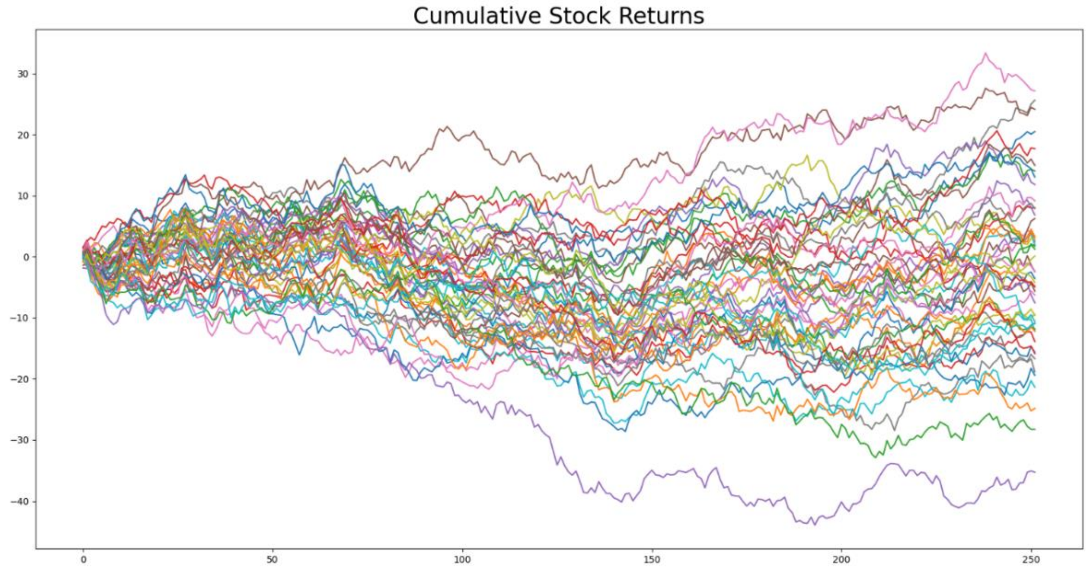

Figure 2: Generated Return Series from the Gaussian Distribution (Without Shocks)

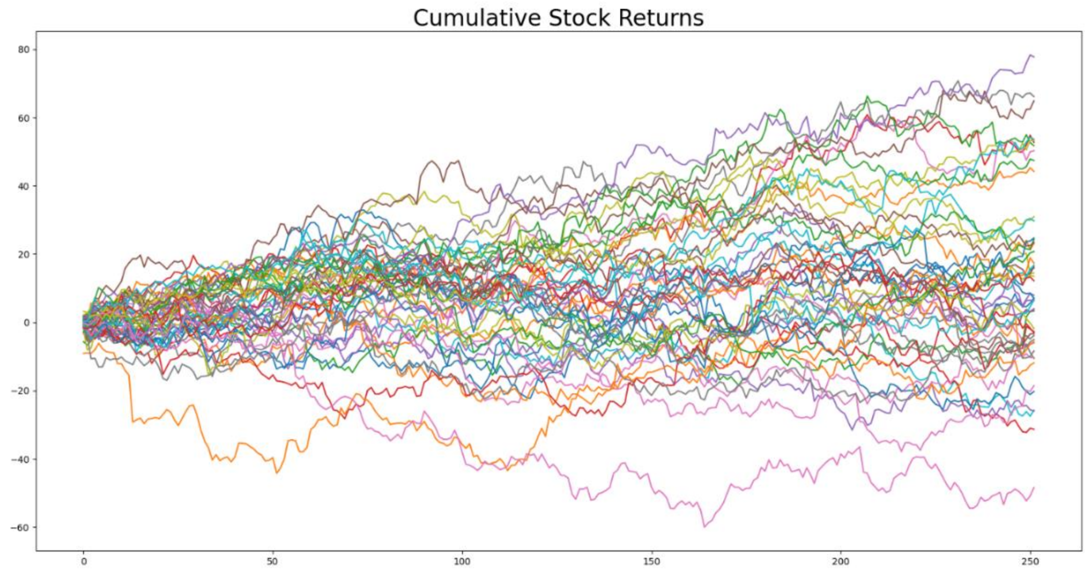

Figure 3: Generated Return Series from the Gaussian Distribution (With Shocks)

For each case, we build two samples of 50×(3*252) observations (three years of daily returns for 50 stocks), where both samples have zero means and same synthetic correlation matrix taken from CorrGAN. The first dataset will be an in-sample dataset used to estimate portfolio allocation weights. The second dataset will be an out-of-sample dataset, having the same parameters as that of in-sample dataset, which will be used to test the stability (measured by the portfolio volatility) of portfolio allocation methods. For each case, we will repeat the above procedure 1000 times, where in each experiment, we will use one of the 1000 correlation matrices generated by CorrGAN. Then we will plot the histogram and average results for each portfolio allocation method.

III. Conclusion and Discussion

1) Research Results

For case 1 (samples from the multivariate Gaussian distribution without shocks), we got the following results:

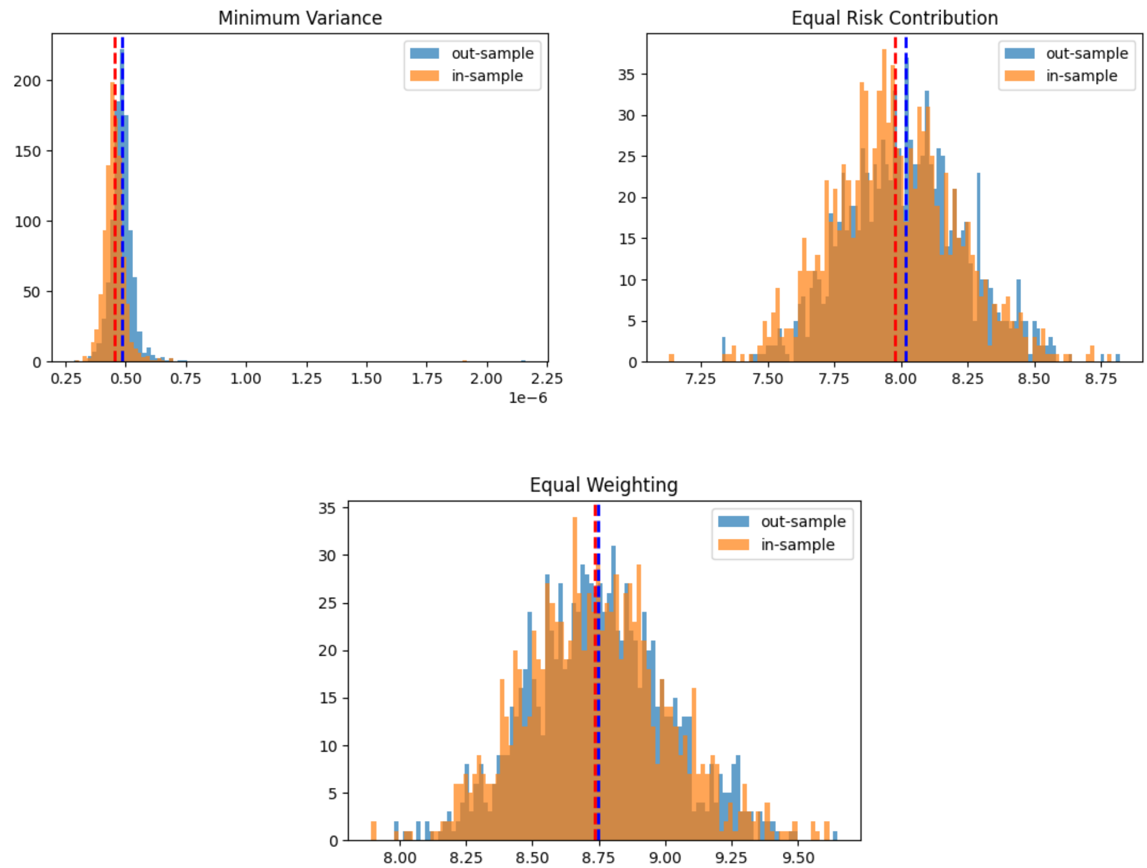

Figure 4: Monte Carlo Experiment Results for the Case Without Shocks (Traditional Portfolio Allocation Methods)

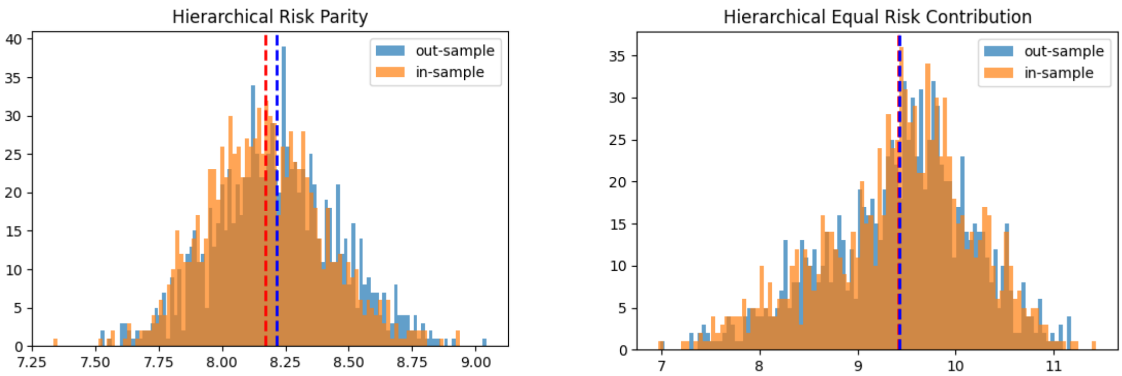

Figure 5: Monte Carlo Experiment Results for the Case Without Shocks (ML-Based Portfolio Allocation Methods)

Based on the above figures, we can see that when no random shocks were applied to the return series, the Minimum Variance portfolio could achieve its goal as the average in-sample and out-of-sample portfolio volatility is almost 0. And also, the ERC portfolio method gave better average volatility results than both ML-based portfolio allocation methods. However, Equal Weighting and HERC methods had the most stable volatilities compared to other methods.

Now, we will plot the results for case 2 (samples from the multivariate Gaussian distribution without shocks), which is closer to reality since financial time series experience jumps in both positive and negative directions.

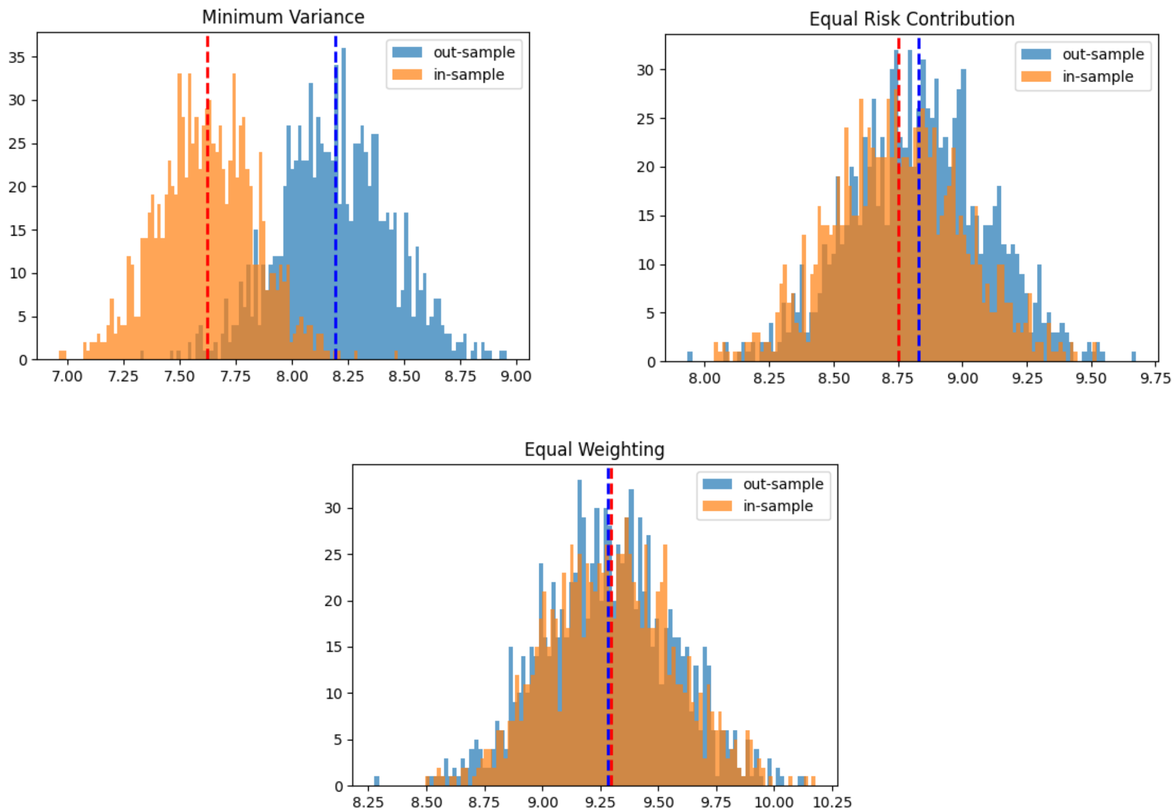

Figure 6: Monte Carlo Experiment Results for the Case with Shocks (Traditional Portfolio Allocation Methods)

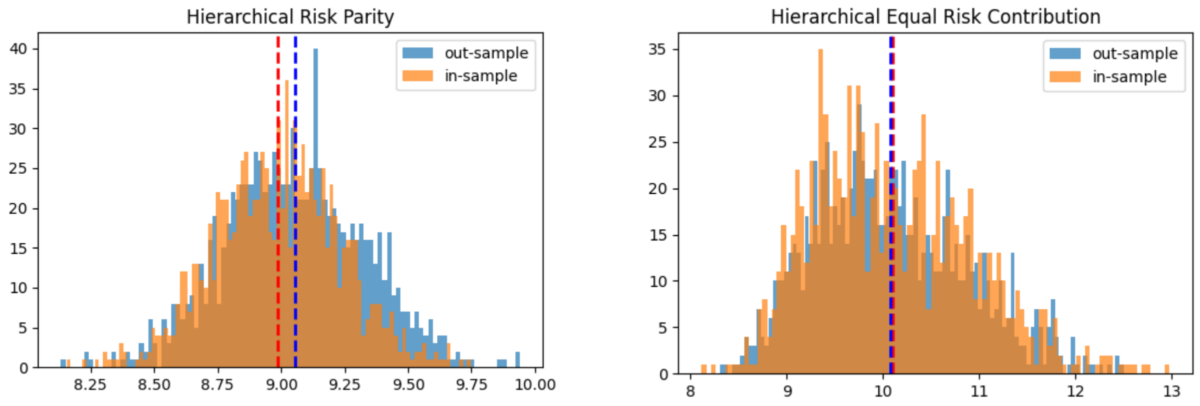

Figure 7: Monte Carlo Experiment Results for the Case with Shocks (ML-Based Portfolio Allocation Methods)

The result of the second case for Minimum variance portfolio is very different compared to the result of the first case. The average in-sample and out-of-sample volatility of Minimum Variance portfolio shifted significantly to the right, and also became more unstable than that of other methods. All other methods were not significantly affected by the shocks and retained stable portfolio volatility.

Clustering based approaching, though much more stable than Minimum Variance method, still could generate lower average volatilities than similar traditional methods such as Equal Risk Contribution method.

2) Discussion

After completing the experiments, we found out that when random shocks are applied to return series, the Minimum variance portfolio has very different average in-sample and out-of-sample volatility compared to the case with no shocks, and the volatility of a portfolio constructed using that method becomes more unstable. Even if the volatility was very unstable in the case of random shocks, the average out-of-sample volatility was still lower than that of all other methods. However, this might change if we apply bigger or more frequent shocks to the return series and also increase the number of experiments.

We initially expected that the ML-based approaches would outperform in terms of average portfolio volatility and stability. However, this turned out not to be the case in our experiments. In the future, in order to have more reliable results for the comparison of traditional and ML-based approaches, we can conduct a larger Monte Carlo experiment by considering more than 50 return series. Also, we can test with different parameters for random shocks, and better approaches to generate realistic correlation matrices or use techniques that directly generate return series resembling real financial time series.

※ References

1. Barziy, I. (2023, February 5). *Corrgan: Realistic Financial Correlation Matrices*. Hudson & Thames. https://hudsonthames.org/corrgan-realistic-financial-correlation-matrices/

2. López de Prado, M. (2016). Building diversified portfolios that outperform out of sample. *The Journal of Portfolio Management*, *42*(4), 59–69. https://doi.org/10.3905/jpm.2016.42.4.059

3. Raffinot, T. (2018). The hierarchical equal risk contribution portfolio. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.3237540

4. *Hierarchical clustering portfolio optimization*¶. Riskfolio. (n.d.). https://riskfolio-lib.readthedocs.io/en/latest/hcportfolio.html#id1

5. Equal risk contribution portfolios - graham capital. (n.d.). https://grahamcapital.com/assets/ResearchPapers/Equal%20Risk%20Contribution%20April%202019.pdf

6. Hierarchical risk parity: How to use Machine Learning for asset allocation? (n.d.-b). https://www.awaleeconsulting.com/wp-content/uploads/2021/03/Machine-Learning-for-Asset-Allocation-AWALEE.pdf

7. Marti, G. (2020). Corrgan: Sampling realistic financial correlation matrices using generative adversarial networks. *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. https://doi.org/10.1109/icassp40776.2020.9053276

8. Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014, June 10). *Generative Adversarial Networks*. arXiv.org. https://arxiv.org/abs/1406.2661

9. Maillard et al. (2010). On the properties of equally weighted risk contributions portfolios. (n.d.-c). http://www.thierry-roncalli.com/download/erc.pdf