



UNIVERSITA' DEGLI STUDI DI PADOVA

**DIPARTIMENTO DI SCIENZE ECONOMICHE ED AZIENDALI
"M.FANNO"**

**CORSO DI LAUREA MAGISTRALE / SPECIALISTICA IN
ECONOMICS AND FINANCE**

TESI DI LAUREA

**"PORTFOLIO ALLOCATION: A COMPARISON BETWEEN
HIERACHICAL RISK PARITY AND MARKOWITZ MODEL
IN PYTHON"**

RELATORE:

CH.MO PROF. BRUNO MARIA PARIGI

CORRELATRICE:

CH.MA PROF.SSA CHIARA DAL BIANCO

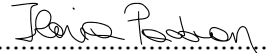
LAUREANDA: ILARIA PADOVAN

MATRICOLA N. 1242492

ANNO ACCADEMICO 2021 – 2022

Dichiaro di aver preso visione del “Regolamento antiplagio” approvato dal Consiglio del Dipartimento di Scienze Economiche e Aziendali e, consapevole delle conseguenze derivanti da dichiarazioni mendaci, dichiaro che il presente lavoro non è già stato sottoposto, in tutto o in parte, per il conseguimento di un titolo accademico in altre Università italiane o straniere. Dichiaro inoltre che tutte le fonti utilizzate per la realizzazione del presente lavoro, inclusi i materiali digitali, sono state correttamente citate nel corpo del testo e nella sezione ‘Riferimenti bibliografici’.

I hereby declare that I have read and understood the “Anti-plagiarism rules and regulations” approved by the Council of the Department of Economics and Management and I am aware of the consequences of making false statements. I declare that this piece of work has not been previously submitted – either fully or partially – for fulfilling the requirements of an academic degree, whether in Italy or abroad. Furthermore, I declare that the references used for this work – including the digital materials – have been appropriately cited and acknowledged in the text and in the section ‘References’.

Firma (signature) 

CONTENTS

ACKNOWLEDGMENTS	9
INTRODUCTION.....	11
1. THE MODEL OF MARKOWITZ	15
1.1 THE MEAN-VARIANCE CRITERION	16
1.2 THE EFFICIENT FRONTIER	17
1.2.1 <i>The Efficient Frontier without Risk-Free</i>	17
1.2.2 <i>The Efficient Frontier with Risk-Free</i>	21
2. THE CRITICAL LINE ALGORITHM AND ITS DRAWBACKS	25
2.1 THE INACCURACY OF THE FORECASTED RETURNS.....	26
2.2 MARKOWITZ'S CURSE	29
2.3 THE LACK OF THE NOTION OF HIERARCHY.....	31
3. THE HIERARCHICAL RISK PARITY MODEL	35
3.1 DESCRIPTION OF THE MODEL	35
3.1.1 <i>Tree Clustering</i>	36
3.1.2 <i>Quasi-diagonalization</i>	40
3.1.3 <i>Recursive Bisection</i>	41
3.2 PERFORMANCE IN-SAMPLE.....	43
3.3 PERFORMANCE OUT-OF-SAMPLE	45
4. EMPIRICAL APPLICATION.....	49
4.1 DATASET	49
4.1.1 <i>Exchange Traded Fund (ETF)</i>	51
4.1.2 <i>The selected ETFs</i>	52
4.1.3 <i>Descriptive statistics of the ETFs</i>	54
4.2 METHODOLOGY	57
4.3 RESULTS	60
4.3.1 <i>Entire Sample</i>	60
4.3.2 <i>Pre-Covid and Post-Covid</i>	67
4.3.3 <i>In-sample versus out-of-sample</i>	71
4.4 CONCLUSIONS	76
CONCLUSION.....	79
APPENDIX	81
A. EFFICIENT FRONTIER FROM MARKOWITZ'S FRAMEWORK.....	81
A.1 <i>Efficient Frontier without risk-free asset</i>	81
A.2 <i>Efficient Frontier with risk-free asset</i>	83

B.	FURTHER EXTENSIONS OF THE HRP MODEL.....	84
C.	PYTHON CODES.....	86
C.1	<i>Source Code for the HRP model.....</i>	86
C.2	<i>Code snippet for the empirical application</i>	88
	REFERENCES	105
	SITOGRAPHY	109

ACKNOWLEDGMENTS

Un immenso grazie

All'Università degli Studi di Padova, per avermi accolta e per avermi dato la possibilità di crescere anche al di fuori dell'ambito accademico grazie alle mille opportunità offerte

Al Ch.mo Prof. Parigi per questa possibilità e la dimostrata passione per i suoi studenti

Alla Ch.ma Prof.ssa Dal Bianco per l'immensa disponibilità, il supporto costante, la guida, i suggerimenti e l'entusiasmo; senza di lei questo lavoro non sarebbe stato possibile

Ai miei genitori, Angelo, Cristina, ai miei fratelli Marco e Maddalena, alle mie nipoti Aurora e Mariasole e ai miei cognati Marika e Francesco, per la presenza costante, l'aiuto continuo e la spinta ad andare sempre avanti

A Francesco per il suo supporto, la sua vicinanza, la sua pazienza e il suo sognare in grande

Ai miei colleghi di corso Francesca, Riccardo, Alessandro, Filippo, Mattia e Gabriele, nei quali ho sempre trovato una mano tesa pronta ad aiutarmi

A Dodo, per la leggerezza e la sua unica ironia, senza le quali l'ambiente universitario non sarebbe stato lo stesso

E infine, alle mie amiche di una vita Martina, Irene, Arianna, Martina e Giulia che hanno sempre creduto in me

INTRODUCTION

The portfolio allocation is a predominant issue in the world of finance; everyday asset managers from all over the world have to elaborate financial portfolios coherent with the objectives of their clients. The most popular model implemented to solve the portfolio allocation problems derives from the Markowitz's framework. The model developed by Markowitz in 1952 is considered to be the foundation of the Modern Portfolio Theory. However, over the years, some limitations related to the model of Markowitz, and more in general, to all quadratic optimizers, have emerged. Contextually, new types of financial allocation have been developed, including the Hierarchical Risk Parity. This model aims to address the concerns of the quadratic optimizers and it has its roots in the growing interest towards artificial intelligence, indeed it requires the application of machine learning techniques.

In the first chapter, the model of Markowitz is presented. He elaborated the "Mean-Variance criterion", according to which the investor seeks to achieve high return and, contemporaneously, low risk. This criterion, together with the principle of diversification, makes possible to identify the set of portfolios ensuring efficient combinations of risk and return. This set of portfolios is called "Efficient Frontier" and it is graphically represented by a hyperbola. The portfolio that delivers the lowest volatility is located at the vertex of the efficient frontier. It is labeled as "Global Minimum Variance" portfolio and it is the only portfolio based exclusively on risk along the efficient frontier.

In the second chapter, the drawbacks of the quadratic optimizers are analyzed, in particular related to the Critical Line Algorithm developed by Markowitz. The forecasted returns, which are used as inputs in those models, can hardly be computed with accuracy and errors in the estimation bring instability in the delivered results. Furthermore, the inversion of the covariance matrix is always required and this step amplifies the effect of the estimation errors in the inputs, especially with the increase of the number of assets considered. This trade-off between diversification and stability of results is known as "Markowitz's curse". The last drawback analyzed is the lack of the notion of hierarchy among the assets, this implies that the weights derived from portfolio allocation can vary with no logic among all investments and this is an additional cause of instability. These limitations can be overcome by implementing the Hierarchical Risk Parity model. The model discards the forecasted returns and it considers only the covariance matrix, adopting a risk parity approach. This kind of approach shifts the focus from the capital allocation to the allocation of risk. Furthermore, the Hierarchical Risk Parity

model avoids the inversion of the covariance matrix by only reorganizing it. Lastly, it forces the weights to vary only among similar assets by implementing the clustering. In this way all causes of instability are removed.

In the third chapter the Hierarchical Risk Parity model is described in detail. As the name implies, it implements a risk parity approach to avoid the instability deriving from the inaccuracy of forecasted returns. All the steps implemented by the algorithm of the model are outlined: it provides the clustering of assets in a tree structure, the quasi-diagonalization of the covariance matrix by reorganizing its elements and the weights allocation within and among clusters according to the Inverse Variance Portfolio. From an analysis conducted by the author of the model, López De Prado, it emerges that the Hierarchical Risk Parity outperforms the minimum variance Critical Line Algorithm portfolio and the Inverse Variance portfolio out-of-sample, in terms of volatility and Sharpe ratio. The analysis is performed on simulated data, through a numerical example and through a Monte Carlo simulation.

In the last chapter, I carried out an empirical application of the Hierarchical Risk Parity model on real data, differently from previous applications implemented on simulated data. I used the programming language Python and I applied machine learning techniques and concepts derived from graph theory. Following the recent trends in financial markets, the model is applied to a portfolio composed of equity ETFs from all over the world. The objective is to evaluate its performance, therefore I compared it with the Global Minimum Variance portfolio from Markowitz's framework. The Global Minimum Variance portfolio has been selected for the comparison given that it is the unique portfolio based exclusively on risk along the efficient frontier, coherent with the fact that the Hierarchical Risk Parity implements a risk parity approach. To test the performance in-depth, two different sets of constraints are imposed on the Global Minimum Variance; one is the positivity constraint and the other is a combination of a lower bound on the ETF tracking the U.S. and an upper bound on the ETFs tracking emerging countries. The analysis is performed in three different settings: on the entire sample from 2011 to 2021, comparing "Pre-Covid" and "Post-Covid" periods and out-of-sample. The analysis out-of-sample is done implementing the "rolling sample" approach with monthly rebalancing of the different portfolios. The results show that Hierarchical Risk Parity outperforms the other models in terms of diversification, stability and, consequently, rebalancing costs. The diversification is assessed by computing the difference in weights with respect to the Equally Weighted portfolio, which is the portfolio optimally diversified. The stability instead is evaluated looking at the frequency and at the scale of changes in the allocation of weights over

time, together with the turnover rate. The Hierarchical Risk Parity model out-of-sample also provides a lower volatility, so a lower risk, with respect to the Markowitz's portfolio with positivity constraint. The main finding corroborates the analyses in previous literature: in facing idiosyncratic and systemic shocks the Hierarchical Risk Parity portfolio does not vary its allocations in an irregular way; the weights do not experience huge changes. It means that when the portfolio faces shocks, the weights are redistributed only among assets belonging to the same cluster and this avoids extreme rebalancing. This also implies that changes in the inputs do not lead to big changes in the output, therefore the intrinsic instability of quadratic optimizers is completely eliminated. However, it provides a volatility only 0.06% lower than the Global Minimum Variance and it presents a slightly lower Sharpe ratio. This could be a consequence of the size or of the correlation structure of the dataset, as explained in the conclusions derived from the application.

1. THE MODEL OF MARKOWITZ

Nowadays, in the finance sector, wealth allocation is still of major importance. Asset managers have the task of creating financial portfolios, considering the various projections in terms of profit and risk. When referring to investment decisions, there exist several different models which can guide the asset manager or the investor in the decision-making process. The aim of portfolio allocation theory is to identify the optimal investment decisions for the investor in a risky environment. The optimal allocation should be coherent with the risk tolerance, the objectives and the investment horizon of the investor.

In 1952, the economist Harry Markowitz developed the most popular model of portfolio allocation in his Ph.D. defense and in 1990 he was awarded with the Nobel Prize in Economic Science together with William F. Sharpe and Merton Miller. Markowitz (1952) described the portfolio selection process and thanks to that he is considered the pioneer of Modern Portfolio Theory. Mangram (2013) defined Modern Portfolio Theory as the process of portfolio construction where the investor seeks to maximize the return and contemporaneously to minimize the risk. According to Markowitz's theory the rational agent in carrying out the portfolio choice acts accordingly to the mean-variance criterion, which will be discussed in detail in the next subsection.

The essential contribution of the aforementioned author is the intuition that each level of risk is associated with a corresponding level of return and the entire set of these combinations forms the so-called "Efficient Frontier". The other important principle identified by Markowitz is the diversification principle which provides, as defined by NASDAQ, that "portfolios of different sorts of assets differently correlated with one another will have negligible unsystematic risk. In other words, unsystematic risks disappear in diversified portfolios, and only systematic risks persist".¹ The information on the correlation among assets allows to achieve a lower incidence of the risk of single assets. Consequently, the agent, which is considered to be rational², is prone to diversify its wealth among different financial assets in order to reach a better risk-return trade-off.

¹ Definition available at the website <<https://www.nasdaq.com/glossary/p/principal-of-diversification>>.

² The agent presents transitive preferences (if he prefers A over B and B over C, for sure he also prefers A over C). This representation is considered as the opposite of a behavioral agent, and it is not always true in reality.

1.1 The Mean-Variance criterion

Markowitz in his dissertation of 1952 tried to identify the optimal allocation process carried out by a rational investor. The author started considering different criteria which could lead the selection process and the consequent allocation of wealth. First of all, the investor cannot aim just to maximize the discounted expected returns, because, in this case, the diversification principle would cease.³ However, even implementing the diversification principle among the securities which maximize the expected return is not enough. This second criterion assumes the existence of a portfolio which guarantees the maximum expected return and the minimum variance and this is not possible given that this kind of diversification cannot remove the entire risk from the portfolio.

The criterion selected is the “Mean-Variance criterion”, also called “Expected returns -Variance of returns (E-V) rule”. Markowitz (1952, p.77) states the famous phrase: the “investor does (or should) consider expected return a desirable thing and variance of return an undesirable thing”. This implies that the investor aims to reach low risk and high return contemporaneously. This is the most complete rule and so it is the one on which the model is based. The E-V rule permits to reach a better diversification, thus a better risk-return trade-off.

Furthermore, the implementation of this rule allows Markowitz’s model to determine not only the portfolio which minimizes the risk subject to a minimum level of expected return, but also the entire set of efficient portfolios. In other words, it identifies all the optimal combinations of risk and return. The measure of risk considered is the variance, while the expected returns are considered through the mean; this is why the optimal portfolio selection problem is formulated via the so-called “Mean-Variance” criterion.

³ The investor could end up allocating the entire wealth to the security with the highest anticipated return.

1.2 The Efficient Frontier

The set of optimal risk-return combinations forms the efficient frontier which is defined by the Corporate Finance Institute as the “set of investment portfolios that are expected to provide the highest returns at a given level of risk. A portfolio is said to be efficient if there is no other portfolio that offers higher returns for a lower or equal amount of risk. Where portfolios are located on the efficient frontier depends on the investor’s degree of risk tolerance.”⁴

There are two cases to analyze: the efficient frontier in the universe composed of $N > 1$ risky assets, without a risk-free asset and the efficient frontier in the universe composed of N risky assets and one risk-free asset with deterministic return r_f .

1.2.1 The Efficient Frontier without Risk-Free

Reporting the notation of Barucci and Fontana (2017), consider the following setting (with no risk-free asset):

- $N > 1$ risky assets yielding random returns $(\tilde{r}_1, \dots, \tilde{r}_N)$;
- $e = (e_1, \dots, e_N)^T$ the vector of expected returns and
- $V \in \mathbb{R}^{N \times N}$ the variance-covariance matrix of the random vector \tilde{r} , which is assumed to be positive definite.

A portfolio contained in the entire set of portfolios Δ_N is defined by the vector $w \in \mathbb{R}^N$ such that $\sum_{n=1}^N w_n = 1$, where w_n is the portion of wealth invested in the n -th asset (the starting wealth is normalized to 1), for $n = 1, \dots, N$.⁵

In the original paper of 1952, Markowitz did not deal with the methodology for the computation of the expected return and variance, but he suggested using the observed data for some periods of the past, substituting theoretical moments with empirical moments. These estimates could be computed using for example sample estimators of mean and variance⁶, exponential smoothing

⁴ The definition is available at the website <<https://corporatefinanceinstitute.com/resources/knowledge/trading-investing/efficient-frontier/>>.

⁵ Further mathematical notations for the setting are reported in the [Appendix A.1](#).

⁶ Also, for example, rolling approaches.

methods⁷, or equilibrium moments obtained for example by estimating factor models (such as the Capital Asset Pricing Model).

There exists a dual representation for identifying the set of efficient portfolios. One possibility is to focus on finding the optimal portfolio $w^* \in \Delta_N$ which for a given expected return level $\mu \in \mathbb{R}$, it minimizes the risk (measured by the variance). The mathematical representation would be

$$\min_{w \in \mathbb{R}^N} w^T V w,$$

under the constraints

$$w^T e = \mu \quad \text{and} \quad w^T \mathbf{1} = 1.$$

In this case, there would be two constraints to consider; the first one relative to the target return μ and the second labeled as the “admissibility constraint”, which guarantees that the entire wealth (no less no more) is allocated.

Another possibility is to find the efficient portfolio that for a given level of risk maximizes the expected return μ . This second representation is reported in [Appendix A.1](#). For the following discussion, the first perspective is taken into consideration.

Given any $\mu \in \mathbb{R}$, the solution to the minimization problem is determined as

$$w^* = g + h\mu.^8$$

As the target return changes, different portfolio solutions are obtained from the minimization problem and the set of all portfolios forms the portfolio frontier.

The portfolio w^* will have variance defined as

$$\sigma^2(\tilde{r}_{w^*}) = \frac{C}{D} \left(\mathbb{E}[\tilde{r}_{w^*}] - \frac{A}{C} \right)^2 + \frac{1}{C} \left(C \mathbb{E}[\tilde{r}_{w^*}]^2 - 2A \mathbb{E}[\tilde{r}_{w^*}] + B \right).$$

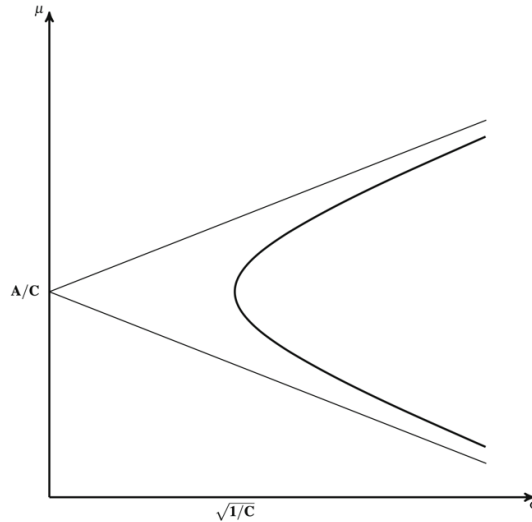
In this way the portfolio’s variance can be expressed as a quadratic function of the target return and, as reported by Barucci and Fontana (2017, p.81), “in the variance-expected return plane $(\sigma^2, \mu) \in (0, +\infty) \times \mathbb{R}$, the portfolio frontier is represented by a parabola with vertex $(1/C, A/C)$.” The graph can be more readable by using the standard deviation σ which has the

⁷ Assigning higher weights to more recent observations and lower weights to further time observations.

⁸ The notations of the scalar quantities are reported in [Appendix A.1](#), together with the computation of the explicit solution.

same unit of return. As Barucci and Fontana (2017, p.81) state, “in the standard deviation - expected return plane $(\sigma, \mu) \in (0, +\infty) \times \mathbb{R}$, the portfolio frontier is represented by a hyperbola centered in the point $(0, A/C)$ and with asymptotes $\mu = A/C \pm \sqrt{D/C}\sigma$.” As a matter of fact, it is common practice to represent the EF in the plane (σ, μ) , where it takes the form of a hyperbola.

Figure 1 - Portfolio frontier in the plane (σ, μ) Data source: Barucci and Fontana (2017, p.81)



The Efficient Portfolio Frontier (EPF), as described by Barucci and Fontana (2017), is the set of portfolios belonging to the part of the portfolio frontier that dominates according to the mean-variance criterion:

$$EPF = \{w^* \in PF \text{ such that } \mathbb{E}[\tilde{r}_{w^*}] \geq A/C\}.$$

As the name says, all the portfolios contained in the efficient frontier are defined as efficient. An efficient portfolio P^* is an admissible portfolio for which the following two conditions hold

- a portfolio \hat{P} with the same expected return and lower risk of P^* does not exist,
- a portfolio \hat{P} with a higher expected return and the same risk of P^* does not exist.

The dominance method provides that in comparing two different portfolios P^* and \hat{P} with different weights, P^* is preferred to \hat{P} (P^* dominates \hat{P}) if

$$\begin{aligned} r_{P^*} &\geq r_{\hat{P}}, \\ \sigma_{P^*} &\leq \sigma_{\hat{P}} \end{aligned}$$

and at least one inequality is strict. It means that for a given level of volatility, higher returns are preferred and for a given level of return volatility, lower risk is preferred. Looking at the equation representing the volatility of the portfolio w^* and at Figure 1, it is straightforward to

see how to a predetermined value σ two different values of μ are associated. Therefore, the efficient frontier corresponds to the upper branch of the hyperbola which dominates the lower one. Furthermore, Figure 1 clearly illustrates the trade-off between expected return and risk, to reach a greater expected return it is necessary to take on more risk.

If at least one inequality is not strict, it is more difficult to identify the optimal investment decision. Different portfolios with different expected return and volatility are not comparable through the dominance method. At this point, to identify the investor's optimal decision, the personal risk aversion coefficient of the investor becomes relevant.

The model's assumptions provide that the rational risk-averse agent wants to maximize his own utility, given the initial wealth and he has free access to all the useful information. Investors follow the E-V rule, taking decisions just according to the first two moments of returns distribution: mean and variance. The process of integrating the preferences of the agent in the optimization problem is reported in [Appendix A.1](#).

The optimal solution depends on the preferences of the agents, so on the risk aversion coefficient η , and on the set of efficient portfolios as a subset of the entire universe of admissible portfolios. The admissible portfolios are all the portfolios that an agent can buy given the initial wealth. Therefore, the optimal investment choice for a rational investor can be found at the tangency between the agent's indifference curves and the efficient frontier.

Along the efficient frontier, there are two specific portfolios that are widely implemented in practice: the Global Minimum Variance portfolio and the Maximum Trade-Off portfolio.

The Global Minimum Variance (GMV or V) portfolio has as objective the minimization of the variance of the portfolio, given the admissibility constraint and without having a predetermined target return. It can be represented as the minimum problem:

$$\min_{w \in \mathbb{R}^N} w^T V w,$$

under the constraint

$$w^T \mathbf{1} = 1.$$

Graphically, the portfolio which allows to reach the minimum variance is located at the vertex of the parabola representing the efficient frontier. As a matter of fact, the GMV portfolio has

expected return $\mathbb{E}[\tilde{r}_{w_{GMV}}] = A/C$ and variance $\sigma^2(\tilde{r}_{w_{GMV}}) = 1/C$. The solution provides the following optimal weights $w_{GMV} = \frac{V^{-1}\mathbf{1}}{\mathbf{1}^T V^{-1}\mathbf{1}}$.

The Maximum Trade-Off (T) portfolio has as objective the maximization of the expected return over the volatility, given the admissibility constraint. The problem can be expressed as:

$$\max_{w \in \mathbb{R}^N} \frac{w^T e}{\sqrt{w^T V w}}$$

under the constraint

$$w^T \mathbf{1} = 1.$$

The solution provides the following optimal weights $w_T = \frac{V^{-1}e}{\mathbf{1}^T V^{-1}e}$ and the corresponding expected return $\mathbb{E}[\tilde{r}_{w_T}] = B/A$ and variance $\sigma^2(\tilde{r}_{w_T}) = B/A$.

The starting objective was to find the optimal investment choice for a rational investor and, again, it can be found at the tangency between the agent's indifference curves and the efficient frontier. The solution is the vector that can be expressed as $w^* = \frac{A}{\eta} w_T - \frac{A-\eta}{\eta} w_{GMV}$; therefore, the optimal weights are a combination of the GMV and the T portfolios. Depending on the risk aversion coefficient η the agent is more prone to invest in one of the two portfolios.⁹

1.2.2 The Efficient Frontier with Risk-Free

The second case to analyze is the efficient frontier when the asset universe is composed of $N + 1$ assets. Reporting the notation of Barucci and Fontana (2017), consider the following setting (with a risk-free asset):

- $N > 1$ risky assets yielding random returns $(\tilde{r}_1, \dots, \tilde{r}_N)$;
- 1 risk-free asset with deterministic return r_f .¹⁰

Given this setting, a portfolio contained in the entire set of portfolios Δ_{N+1} is defined by the vector $w = (w_1, \dots, w_n)^T \in \mathbb{R}^N$, where w_n is the portion of wealth invested in the n -th risky

⁹ Further passages and implications can be found in the [Appendix A.1](#).

¹⁰ In [Appendix A.2](#) it is possible to find further characteristics of the setting.

asset, for $n = 1, \dots, N$, and $1 - \sum_{n=1}^N w_n$ ¹¹ is the portion of wealth invested in the risk-free asset.

In this scenario, the formulation of the minimization problem for a given expected return level $\mu \in \mathbb{R}$ is:

$$\min_{w \in \mathbb{R}^N} w^T V w,$$

under the constraint

$$w^T e + (1 - w^T \mathbf{1}) r_f = \mu.$$

In this case, the admissibility constraint is embedded in the μ expression: the sum of w^T and $(1 - w^T \mathbf{1})$ is equal to 1.

By considering the initial universe of N risky assets and adding one asset to it, the universe of combinations of risk and return enlarges. The efficient frontier moves to the left, reaching new investment opportunities presenting lower volatility and higher return, thanks to the diversification effect. Furthermore, in this case, it is possible to allocate the entire wealth in the risk-free asset reaching zero volatility.

Given any $\mu \in \mathbb{R}$, the solution to the minimization problem is determined as

$$w^* = \frac{\mu - r_f}{K} V^{-1} (e - r_f \mathbf{1}).$$
¹²

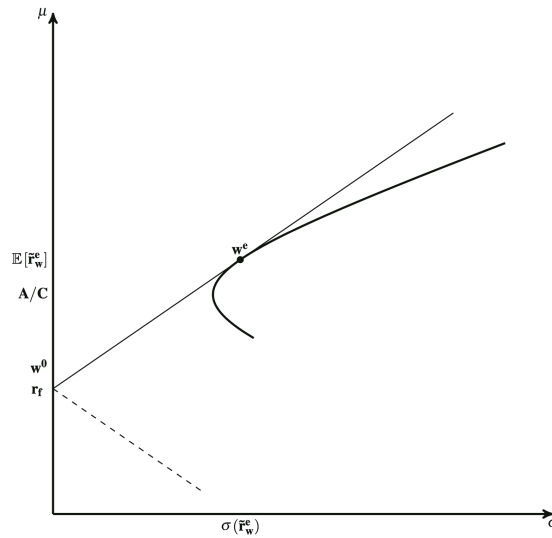
As the target return changes, different portfolio solutions are obtained from the minimization problem and the set of all portfolios forms the portfolio frontier.

As reported by Barucci and Fontana (2017, p.89), “the portfolio frontier PF* can be represented in the standard deviation-expected return plane by two half-lines with slope \sqrt{K} and $-\sqrt{K}$, respectively, originating from the point $(0, r_f)$the half-line with positive slope represents the efficient part of the portfolio frontier (EPF*)”. The portfolios with return lower than r_f are dominated by an investment in r_f , indeed in the plane considered by standard practice (σ, μ) , moving along the straight line towards right means to allocate some wealth in the risky assets.

¹¹ This also represents the opportunity to invest or to borrow, according to the sign.

¹² Further explanations are reported in the [Appendix A.2](#).

Figure 2 - Portfolio frontier with a risk-free asset ($A/C > r_f$) Data source: Barucci and Fontana (2017, p.92)



In this setting, a relevant portfolio to identify is the Tangency Portfolio or Maximum-Sharpe Portfolio. The Sharpe ratio is a measure of portfolio's performance. According to Barucci and Fontana (2017, p.93), "given a portfolio $w \in \mathbb{R}^N$, the corresponding Sharpe ratio $SR(w)$ is defined as follows:

$$SR(w) := \frac{\mathbb{E}[\tilde{r}_w] - r_f}{\sigma(\tilde{r}_w)}$$

i.e., the ratio between the return risk premium of a portfolio and the standard deviation of the associated random return. In the standard deviation – expected return plane, the Sharpe ratio represents the slope of the straight line which connects the point $(0, r_f)$ to the point $(\sigma(\tilde{r}_w), \mathbb{E}[\tilde{r}_w])$ ". The portfolio weight is $w_T = \frac{V^{-1}(e - \mathbf{1}r_f)}{\mathbf{1}^T V^{-1}(e - \mathbf{1}r_f)}$.

By maximizing the utility function reported in [Appendix A.2](#), the optimal weights for the risky assets are

$$w = \frac{1}{\eta} V^{-1}(e - \mathbf{1}r_f)^T = \frac{A - Cr_f}{\eta} w_T.$$

The wealth allocated in the risky assets is directly related to the tangency portfolio and, as a result, the optimal portfolio is composed of an investment in the risk-free asset and in the tangency portfolio. Given that the optimal choice corresponds to the tangency of the efficient frontier with the investor's indifference curves, the agent decides to allocate more wealth towards one of the two constituents of the portfolio depending on the risk aversion coefficient.

2. THE CRITICAL LINE ALGORITHM AND ITS DRAWBACKS

The traditional Markowitz model previously described is the most popular model among all the quadratic optimizers. By the way, Markowitz in 1954, while working for RAND Corporation, developed the Critical Line Algorithm (CLA). With respect to other generic quadratic programming methods, it guarantees to find the precise correct solution and even to have a stopping time. As described by López De Prado (2016) the CLA is a quadratic optimizer developed for constrained optimization problems with inequality constraints and it avoids the application of Karush-Kuhn-Tucker conditions thanks to the concept of “turning point”.¹³

In practical financial allocation, it is common to face optimization problems that include inequality constraints, such as a lower and an upper limit for each weight of the portfolio. Since the CLA is the proper algorithm for solving this type of problems, there should exist several implementations of it, however, this is not the case. As recognized by Bailey and López De Prado (2013), asset managers and, more in general, financial firms use generic optimizers which are not tailor-made for optimization problems with inequality constraints, frequently delivering suboptimal allocations. This is a consequence of the absence of a published proper algorithmic implementation of the CLA. The only implementation code existent is the one published by Markowitz and Todd (2000), which is developed in Excel’s Microsoft Visual Basic for Applications with some consequent inefficiencies.¹⁴ Bailey and López De Prado (2013) provide an implementation of the CLA accessible to everyone in a proper scientific language. The authors choose to create a Python class object which is free and of simple readability.

However, the CLA algorithm presents some drawbacks. The limitations of the model, which will be analyzed in-depth in the following subsections, are related to the inaccuracy of the forecasted returns, the instability deriving from the inversion of the matrix (Markowitz’s curse) and from the lack of a hierarchical structure.

¹³ Further details can be found in Bailey and López De Prado (2013).

¹⁴ The inefficiencies of VBA-Excel are linked to the low computational performance, the limit to use covariance matrix with a maximum size of (256x256) and the necessity to manually adjust the spreadsheet for each different problem.

2.1 The inaccuracy of the forecasted returns

According to Michaud (1998, p.5), the most significant practical limitation of Markowitz's model is the instability: "small changes in input assumptions often imply large changes in the optimized portfolio." The problem is that the computational process of the efficient frontier makes an excessive use of inputs which are statistically estimated, such as the mean of forecasted returns; consequently, in the optimized portfolios, the effects of estimation errors are maximized. Hence, the first drawback comes from the inaccuracy of the forecasted returns: returns can hardly be predicted with adequate precision because some uncertainty is always implied when predicting the future. As all quadratic optimizers, even the CLA is characterized by an oversensitivity to estimation errors, according to López De Prado (2016), small errors in the forecasted returns can deliver drastically different efficient frontiers and, so, really different allocations.

To improve the reliability of the investment value of the generated portfolios, it is important to overcome this first drawback. In the last years, various authors started to focus exclusively on the covariance matrix by discarding the forecasted returns because, as recognized by Choueifaty and Coignard (2008), variance can be assessed with sufficient confidence, while returns are more difficult to estimate with sufficient accuracy. A consequence is the increasingly widespread implementation of "risk-based" asset allocation, also called "risk control" technique or "risk parity" approach. This new type of approach has as objective the allocation of the risk instead of the traditional allocation of capital.

Furthermore, historically, the portfolio allocation was based on the distribution of wealth among different types of assets, but as reported by Jurczenko (2015), after the global financial crisis of 2007-2009 it was clear that the financial investors should start shifting more attention towards the risk embedded in their portfolios. As recognized by Dalio, Jensen and Prince (2015, p.2), this comes from the fact that the "traditional institutional portfolio has about 60% of its money in stocks and 40% in bonds." Consequently, as Allen (2010) clarified, the total volatility of the traditional institutional portfolio derives for the 70% to the 85% from the stocks. This comes from the fact that stocks are more volatile than bonds and more than half of the wealth is allocated to them. This kind of portfolio composition delivers a risk structure completely centered on stocks, leading to underdiversification. Focusing on risk allows a more meaningful diversification, indeed a study performed during those years by Choueifaty and Coignard

(2008) showed how traditional methodologies are dominated by risk-efficient portfolios in various aspects.

The risk parity approach, as intended by most investors, has as objective the creation of a portfolio where the different assets contribute in the same way to the risk of the entire portfolio. To bring at the same level the impact of the volatility of stocks and bonds, it is particularly useful to leverage the bonds. Even if it is not a prerogative of this approach, the majority of risk parity investors makes a large use of leveraging. According to Dalio, Jensen and Prince (2015), leveraging the bonds to match the volatilities of the stocks, would increase, of course, the expected risk of bonds, but also their expected return. The expected return would increase because if bonds are acquired borrowing cash, the profit is greater given that the return of bonds is higher with respect to the return of cash. In the portfolio, the higher portion of risk coming from bonds allows to reduce the portfolio risk. These consequences make the leveraging useful for reaching a better risk-return ratio. If no leveraging is implemented, the portion invested in stocks would be reduced and spread in bonds, but by doing so, the expected return of the portfolio would be reduced. Initially this approach was focused only on the institutional portfolio of stocks and bonds, but nowadays risk parity is considered as a real investment criterion for the entire universe of investment. Disregarding the forecasted returns and controlling only the risk allows to find adequate portfolios.

The risk parity approach includes a variety of allocation techniques. The most popular risk-based allocation methodologies are the Minimum Variance, the Equal-Weighting (1/N), the Maximum Diversification, the Inverse Volatility (IV) and the Equal Risk Contribution. To analyze some of these techniques it is essential to define the weight per asset class as the sum of the weights of each asset belonging to the class and the risk allocation per asset as the marginal contribution to risk (MCR) for each asset. The MCR represents the effect on the portfolio volatility given a change in the weight of the asset. Referring to the mathematical notation proposed by Jurczenko (2015), it corresponds to the partial derivative of the volatility of the portfolio with respect to the weight of each asset: $MCR_t^i = \frac{\partial \sigma_t^P}{\partial w_t^i}$.

The most widely used technique is the Equally Weighted Risk Contribution (ERC) where the objective is to create a portfolio where the contribution to the total portfolio volatility is exactly the same for each asset class. In mathematical terms, it can be represented as

$$w_t^{i,RP} \cdot MCR_t^i = constant, \forall i ;$$

where the constant is equal to the $\frac{1}{N_t}$ of the portfolio volatility. This mathematical representation does not include the possibility to go short on some constituents of the portfolio, but it generates a long-only type of allocation. To implement a risk-parity long-short portfolio, signed correlations should be used.¹⁵

Another popular technique is the Inverse Volatility, as Jurczenko stated (2015, p.18) to implement it means to “set each weight proportional to the stand-alone volatility of the corresponding asset and next normalize so that the weights sum to unity. This volatility-weighting in the cross-section yields: $w_i = \frac{\frac{1}{\sigma_i}}{\sum_j \frac{1}{\sigma_j}}$ ”. As the denomination of this technique suggests, it assigns lower weights to the most volatile assets.

Along with the most popular techniques, also the recently developed Hierarchical Risk Parity portfolio adopts the risk parity approach. In the process of definition of the optimal allocations of weights, more precisely in the recursive bisection stage, the Inverse Volatility technique is implemented as reported in [Section 3](#). In this way, the portfolio focuses only on risk, so only on the covariance matrix, overcoming the instability issue related to the inaccuracy of the forecasted return.

¹⁵To identify the assets on which go short, the sign associated to the variance must be reintroduced after the computation of the correlation.

2.2 Markowitz's Curse

In the next paragraphs the second important limitation of the CLA, the so-called “Markowitz’s Curse”, will be analyzed. This is an important issue which leads to instability in the model’s results. When the covariance matrix is ill-conditioned it means that a small deviation in the inputs can determine big changes in the output. In this specific case, it is highly probable to have deviations in the inputs given that the inputs are the expected returns which are difficult to forecast with accuracy. Moreover, the CLA model requires the inversion of the covariance matrix that, in such a case of ill-conditioned matrix, can entail significant errors. This set of causes leads the model to deliver very unstable results.

In the quadratic-programming methods, it is always required the inversion of a positive-definite covariance matrix¹⁶ and, according to Bailey and López Del Prado (2012), the inversion of an ill-conditioned matrix delivers highly unstable results. Belsey, Kuh and Welsch (2004) define that as the condition number increases, the ill-conditioned being of the matrix increases. In numerical analysis, the condition number of a matrix measures the sensitivity of the output to perturbations in the input data; in other words, it represents the extent to which the result of a function can change depending on changes or errors in the input argument. Therefore, the ill-conditioned nature of the matrix may severely impact the accuracy of the computed solution. Mathematically, for a covariance matrix, López De Prado (2016) clarifies that the condition number is the module of the maximal eigenvalue over the module of the minimal eigenvalue.

From an analysis conducted by López De Prado (2016), it is possible to see how the condition number increases as the correlation becomes stronger within the matrix itself. The identity matrix¹⁷, which can be associated to a diagonal correlation matrix, has a condition number equal to 1. Condition numbers close to 1 suggest that the matrix is well-conditioned, this implies that the inverse matrix can be calculated with sufficient precision. When the correlation becomes stronger within the matrix itself, the need for diversification increases. Therefore, new assets have to be added to the investment universe, but the greater the number of assets, the more likely they will be correlated. As shown by López De Prado (2016) analysis, if new correlated

¹⁶ It means that all the eigenvalues must be positive.

¹⁷ The identity matrix is a matrix that presents only numbers 1 along the diagonal and 0 in all the remaining positions.

investments are added to the matrix, so new correlations are introduced in the matrix outside the diagonal, the maximal eigenvalue increases while the minimal decreases and this leads to a greater condition number. A higher condition number implies that also very small numerical errors can lead to different solutions, conducting to unstable inverse matrices. As a matter of fact, the singular matrix presents the highest condition number in the analysis and its inverse could not even be computed. This is the “Markowitz’s curse”: with the increase of correlation among the assets of the portfolio it is necessary to improve the diversification, but at the same time this leads to a higher probability to achieve unstable results. Therefore, the effect of estimation errors erases the perks of diversification. Summing up, the Markowitz’s curse could be interpreted as this trade-off between the diversification need and the stability of the results.

Furthermore, there is another factor which contributes, to a lesser extent, to the inaccuracy and instability of the results. It comes from the fact that the CLA model requires the inversion of the covariance matrix and this operation is not feasible for a singular matrix. This is an issue because in order to compute a not singular covariance matrix, at least $\frac{1}{2}N(N + 1)$ independent and identically distributed (IID) observations are needed, if the matrix in question has size N . The problem is that the underlying correlation structure can change significantly considering such long periods of time. Furthermore, if the size of the covariance matrix increases each element is estimated with fewer degrees of freedom, so with less precision. This combination of issues brings even more instability in the model.

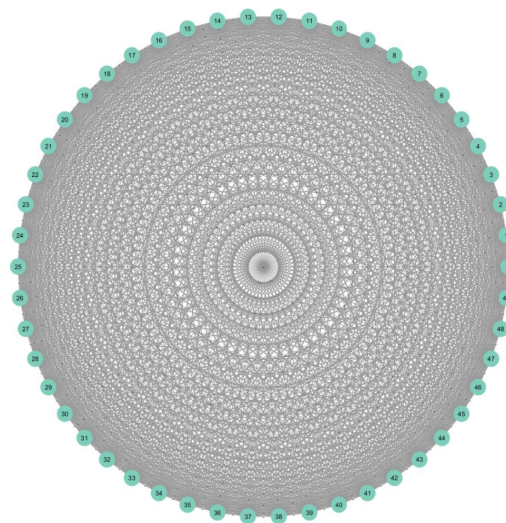
The author López De Prado developed a new model, labeled “Hierarchical Risk Parity” (HRP), where these issues are addressed. In the HRP model the inversion of the covariance matrix is not required; the passage is avoided thanks to the implementation of hierarchical clustering of the assets and thanks to the quasi-diagonalization of the covariance matrix. In this way, it is possible to avoid what is defined as the “Markowitz’s curse”: since the matrix inversion is not necessary, the HRP can calculate financial portfolio’s weights even starting from singular or ill-degenerated covariance matrices. Further details will be provided in [Section 3](#) along with the description of the Hierarchical Risk Parity model.

2.3 The lack of the notion of hierarchy

To overcome the instability pitfall of the variety of quadratic optimizers and, in particular of the CLA, a lot of different authors presents extensions trying to reach robustness. All the proposed modifications rely on the traditional scope of mathematics, in particular on geometry, linear algebra and calculus; such as the addition of further constraints, the implementation of the Bayesian approach or the use of a shrinkage estimator for the covariance matrix. However, there exist also other approaches which can help overcome the limitations of the CLA model, for instance through notions derived from graph theory.

One of the CLA's drawbacks arises when considering the correlation matrix in the vector space composed by the returns. The matrix of correlations is a linear algebra object. According to López De Prado (2016, p. 4) it can be considered a measure of “the cosines of the angles between any two vectors in the vector space formed by the returns series”. The issue stands in the fact that the vector space composed by returns is designed as a fully connected graph, where each vertex (node) is connected to all remaining vertices (nodes). This kind of structure is also labeled “complete graph” and the consequence is that there are no logical relations among the different nodes, but every single node can be replaced by another. This is a major source of instability.

Figure 3 - Complete (fully connected) graph Data source: López De Prado (2016, p.20)



In Figure 3, the complete (fully connected) graph of a covariance matrix of size 50×50 is represented. There are 50 nodes which correspond to the correlations that lie along the diagonal of the correlation matrix and 1225 edges. The 1225 edges correspond to all the partial correlations which lie outside the diagonal contained in the matrix.¹⁸ When the computation of the inverse matrix is carried out, all the partial correlations must be considered. In this way every little error in the estimation is amplified. To reduce this effect, could be useful to not consider all the partial correlations, but only the relevant ones. A solution could be to eliminate all superfluous edges by implementing a hierarchical classification of them.

The main problem is that in the fully connected graph there is no hierarchical structure and so each investment can be considered a good substitute for each other investment. This has a direct effect on the weights: given that there is no distinction among different investments, the optimal weights could end up shifting with no criteria among each asset, giving rise to additional instability. Small errors in the inputs could lead to totally different allocations. The perspective to introduce provides that logical meaningful relationships among the investments can be identified. The similarity or dissimilarity among investments allows to create groups of peers where assets are valid substitutes of each other. For instance, the assets could be grouped according to different criteria: asset class, geographical area, sector or size. Then, the assets belonging to a specific cluster compete for wealth allocation. The meaning is that in the universe of assets considered for the investment, there are assets which are very “far” from each other and so their partial correlations may not be considered because of their irrelevance. Grouping together similar investments implies that the weights do not vary in unintended ways among all assets, but only within a specific group.

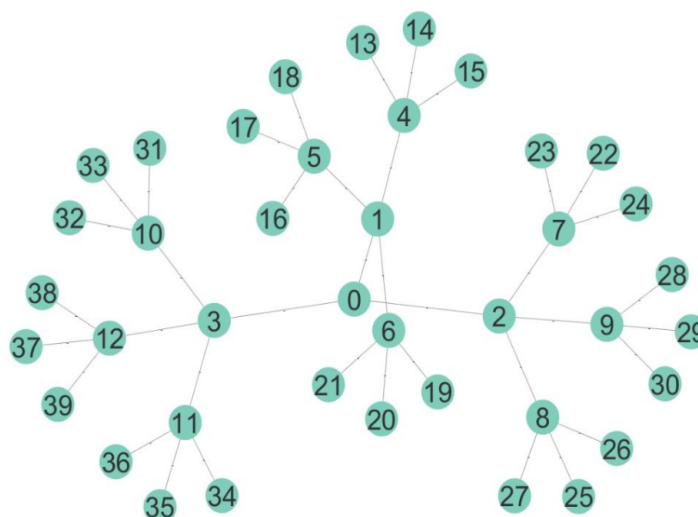
To prevent the unintended variations of weights, it is necessary to introduce a hierarchical structure. A possible option is to implement a tree structure which allows to drop the inessentials edges, in fact, it presents N vertices, denominated “nodes”, connected by $N-1$ links, denominated “edges”. Looking at the tree structure in Figure 4, it is intuitively visible that the weights are allowed to vary only among the assets contained in a single cluster and this applies to each level determined by the tree structure. In this way, the hierarchical structure leads to

¹⁸ The correlation matrix contains 50×50 elements, so 2500 correlations. The 50 correlations on the diagonal are the correlations of each asset with itself, while the remaining 2450 are the partial correlations of each asset with another. The matrix is symmetric: only the upper or the lower triangle must be considered to avoid the double-count of the partial correlations. In this way only 1225 partial correlations are considered as edges.

more stable solutions with respect to the geometrical relationship previously described with the fully connected graph. Another advantage of the hierarchical clustering technique is the coherence with the allocation process currently adopted by asset managers. Thanks to the hierarchical structure, the allocation of weights is done in a top-down way, exactly as the standard practice for portfolio allocation.¹⁹

The Hierarchical Risk Parity portfolio implements the tree clustering as its first step in the definition of the optimal allocation of weights. Further details will be provided in [Section 3](#) along with the description of the Hierarchical Risk Parity model.

Figure 4 - Tree structure Data source: López De Prado (2016, p.20)



Summing up, the quadratic optimizers deliver unstable results. This instability comes from multiple sources. The first analyzed source is the inaccuracy of forecasted returns, which can be addressed by implementing a risk parity approach. The second source is the Markowitz’s curse, which originates from the inversion of the covariance matrix, and it can be tackled by avoiding the inversion of the matrix by applying alternative steps. The third source of instability is the lack of hierarchy among assets, which allows illogical variations in weights. This can be overcome through graph theory, for example, adopting a tree structure to implement hierarchical clustering. There is a model which incorporates all these listed solutions and it is the Hierarchical Risk Parity.

¹⁹ As highlighted by López De Prado (2016, p.5) the standard practice provides allocations from “asset class to sectors to individual securities”.

3. THE HIERARCHICAL RISK PARITY MODEL

During the last years, a new model for portfolio allocation has been developed by López de Prado (2016).²⁰ The model in question is the Hierarchical Risk Parity (HRP) model, which merges together graph theory and unsupervised machine learning techniques, exploiting the modern mathematics knowledge.

The HRP allows to reach a greater stability of results, overcoming the drawbacks of quadratic optimizers, such as concentration and instability. The HRP does not require the inversion of the covariance matrix because of the implementation of the hierarchical clustering and of a passage that provides the reorganization of the elements contained in the covariance matrix, without inverting it. Henceforth, through this model is possible to perform the portfolio allocation, even starting from singular or non-positive definite matrices.

The algorithmic implementation of the HRP model must be carried out through three different stages: the tree clustering, the quasi-diagonalization and the recursive bisection. In the following subsections, each stage will be presented in detail. The mathematical description of the HRP algorithm follows step by step the model and the mathematical notation of López De Prado (2016) and it is carried out in deterministic logarithmic time $T(n) = O(\log_2 n)$.²¹

3.1 Description of the model

The first stage of tree clustering implies the identification of clusters composed of similar investments, addressing the concerns related to the fully connected graph and the lack of hierarchy. The second stage of quasi-diagonalization involves the reorganization of the covariance matrix, with no need for inversion. The last stage of recursive bisection allocates the weights to each cluster and to each asset according to the inverse variance allocation technique.

²⁰ Further information on the author is reported on the website <<https://www.quantresearch.org>>.

²¹ This setup is known in computer science for achieving lower algorithmic time-complexity when dealing with large datasets. More specifically the \log_2 indicates the binary search structure.

3.1.1 Tree Clustering

The first stage for the algorithmic implementation of the HRP model is the tree clustering. First of all, clustering is an unsupervised machine learning technique used to identify and group together similar objects, within a dataset. The classification in groups is carried out according to a specific similarity measure. In general, unsupervised machine learning applies algorithms which are self-sufficient in identifying similarities and differences for clusters' creation and underlying structures in the dataset. This technique is defined as "unsupervised" because it does not require the supervision of the user.²²

As explained before, the hierarchical clustering, and more specifically the tree structure, allows to achieve greater stability. The similar assets contained in a cluster compete for the allocation of capital, quoting López De Prado (2016, p.4), in allocating the wealth to "a large publicly-traded U.S. financial stock like J.P. Morgan, ... [the choice should fall on] adding or reducing the allocation to another large publicly-traded U.S. bank like Goldman Sachs, rather than a small community bank in Switzerland, or a real estate holding in the Caribbean."

Regarding the hierarchical clustering, there are two different approaches: the divisive analysis (DIANA) and the agglomerative nesting (AGNES). As explained by Pfitzinger and Katzke (2019), on one hand, the DIANA involves a top-down approach where the universe of objects is contained in a single cluster and then it is iteratively divided into sub-clusters, till the point that each asset is a cluster itself. On the other hand, the AGNES involves a bottom-up approach where the closest objects and, subsequently, clusters are iteratively merged into bigger clusters.

The similarity between clusters is evaluated according to the linkage criterion selected. The AGNES approach linkage criteria comprehend:

- the Simple or Single Linkage, where the minimum distance between any two points in the clusters is considered;
- the Average Linkage (AL), where instead the average distance is considered;
- the Complete Linkage (CL), where the maximum distance is considered and
- the Ward's method (WM) where the distance between two clusters is based on the effect on the sum of squared error when the clusters are merged.

²² The exact definition is available at the website of International Business Machines (IBM) <<https://www.ibm.com/cloud/learn/unsupervised-learning>>.

For his analysis, López De Prado selected the AGNES single-linkage hierarchical clustering, identifying the correlation matrix as the starting point for the construction of the similarity measure.

The model starts by taking into consideration the observations of time series of returns for N variables, representing N assets, over T periods, obtaining the matrix of returns X of size $T \times N$. The objective is to merge the N assets columns to extract the tree structure from the data. As a starting point, the computation of the correlation matrix of returns, with size $N \times N$ must be carried out. Defining $\rho_{i,j}$ as the correlation coefficient between assets i and j , all the correlations among the different variables N form the correlation matrix. Henceforth, $\rho = \{\rho_{i,j}\}_{i,j=1,\dots,N}$ is the correlation matrix, where $\rho_{i,j} = [X_i, X_j]$. The construction of the tree graph can be described through six distinct steps.

Firstly, from the correlation matrix, the distance matrix is derived by using a correlation-based metric. The distance metric is defined making use of the Pearson's correlation definition, the formula of the Euclidean distance and the formula for the standardization of vectors.²³ As López De Prado (2016, p.5) states, the distance measure is defined as “ $d : (X_i, X_j) \subset B \rightarrow \mathbb{R} \in [0,1]$,”

$$d_{i,j} = d [X_i, X_j] = \sqrt{\frac{1}{2}(1 - \rho_{i,j})},$$

where B is the Cartesian product of items in $\{1, \dots, i, \dots, N\}$ ”. The distance metric is defined between columns and it is applied to the correlation matrix, as a result the distance matrix $D = \{d_{i,j}\}_{i,j=1,\dots,N}$ of size $N \times N$ is obtained. This can be considered an appropriate metric space since it meets the requirements of non-negativity, coincidence, symmetry and sub-additivity.²⁴

Secondly, the computation of the Euclidean distance of the correlation distances is computed. As described by López De Prado (2016, p.5), it is applied “between any two column-vectors of D , $\tilde{d}: (D_i, D_j) \subset B \rightarrow \mathbb{R} \in [0, \sqrt{N}]$,”

²³ The detailed proof for the definition of the correlation-based metric is reported in López De Prado (2016).

²⁴ The non-negativity comes from $d[X, Y] \geq 0$, the coincidence means that $d[X, Y] = 0 \Leftrightarrow X = Y$, the symmetry stands in the fact that $d[X, Y] = d[Y, X]$ and the sub-additivity is derived from $d[X, Z] \leq d[X, Y] + d[Y, Z]$. The detailed proof it is again reported in López De Prado (2016).

$$\tilde{d}_{i,j} = \tilde{d}[D_i, D_j] = \sqrt{\sum_{n=1}^N (d_{n,i} - d_{n,j})^2}.$$

This $\tilde{d}_{i,j}$ is a distance function calculated on columns of the distance matrix and so on the whole correlation matrix, not exclusively on a specific cross-correlation pair.

Thirdly, there is the main step of clustering the assets. Pair of columns are clustered together such that the columns $(i^*, j^*) = \operatorname{argmin}_{(i,j)_{i \neq j}} \{\tilde{d}_{i,j}\}$ and the resulting cluster is labeled as $u[1]$ and it is equal to (i^*, j^*) .

Fourthly, the distance of the $u[1]$ cluster from the remaining assets of the matrix must be identified. At this point, the linkage criterion selected becomes relevant in defining the distance. The Single Linkage criterion is implemented, so, the item i belonging to the matrix \tilde{d} should be $\dot{d}_{i,u[1]} = \min \left[\{\tilde{d}_{i,j}\}_{j \in \mu[1]} \right]$ far from the cluster $u[1]$, considering the minimum distance. In this way it is possible to update the matrix $\{\tilde{d}_{i,j}\}$ with the new cluster u .

Fifthly, the newly calculated $\dot{d}_{i,u[1]}$ is appended to $\{\tilde{d}_{i,j}\}$ and the columns and rows $j \in u[1]$ that have been clustered are cutted off.

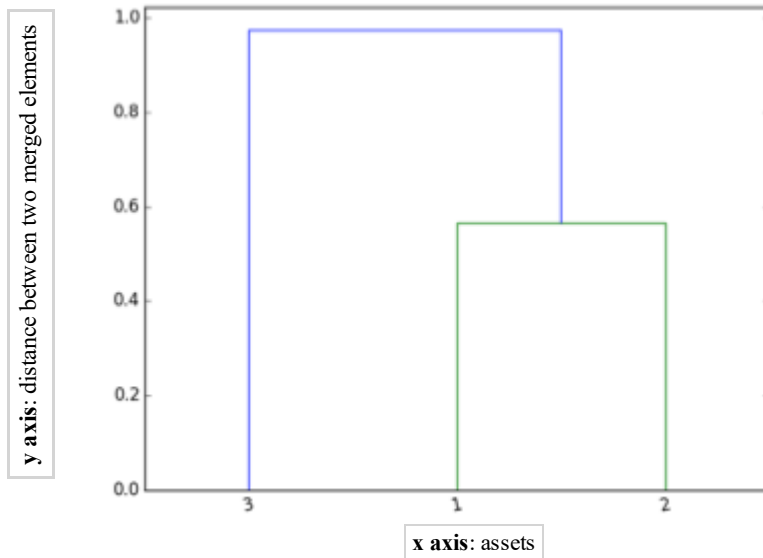
Sixthly, the process of clustering items and updating the distance matrix is iterated, so from the third to the fifth step, until $N-1$ clusters are appended to matrix D . Thereby, at the last step, there will be only one single cluster which contains the entire universe of original assets.

In this way, a series of characteristics is associated to each cluster m . The various steps deliver a linkage matrix $(N - 1) \times 4$, where $(N - 1)$ is the number of cluster and 4 represent the 4-tuple associated to each cluster. The tuple is $Y = \{(y_{m,1}, y_{m,2}, y_{m,3}, y_{m,4})\}_{m=1, \dots, N-1}$, where $(y_{m,1}, y_{m,2})$ are the elements forming the cluster. The component $y_{m,3}$ coincides with the distance between the two elements, which corresponds to $\tilde{d}_{y_{m,1}, y_{m,2}}$; while $y_{m,4}$ corresponds to the original number of elements contained in m .

The hierarchical structure derived from this first stage of the model is a tree structure and it can be plotted as a dendrogram graph. In Figure 5 there is the visual representation of the formed clusters and on the vertical axis y is reported the distance \tilde{d}_{i^*, j^*} at which the merge of the different clusters took place.

Figure 5 - Dendrogram of cluster formation

Data source: López De Prado (2016, p.21)



To put it more precisely, the result of the tree clustering is a Minimum Spanning Tree. Raffinot (2018) explains that a Minimum Spanning Tree is widely used in financial applications. Its focus is to identify the relevant and meaningful links among assets, in order to discard the irrelevant ones. As explained before, considering N asset delivers a structure with only $N - 1$ relevant edges instead of the $\frac{1}{2}N(N - 1)$ delivered by the complete graph.

3.1.2 Quasi-diagonalization

The second stage for the algorithmic implementation of the HRP model is the quasi-diagonalization. In this step, the principal improvement of the model with respect to the Markowitz model and to Principal Component Analysis (PCA) or similar procedures is done. Instead of executing the inversion of the matrix, the rows and the columns in the covariance matrix are only reordered by using the information provided from the first stage of hierarchical clustering. The quasi-diagonalization does not require the change of the basis but, simply, places the greatest values, so the highest correlations, along the diagonal of the matrix. In this way, the more correlated investments are situated in closer positions, while the less correlated are situated in distant positions. This step takes the name of quasi-diagonalization because putting the greatest values on the diagonal makes the matrix similar to a diagonal matrix, where the correlations outside the diagonal correspond to 0 and those on the diagonal to 1. The closer investments are the most similar, while the farthest are the most dissimilar. Furthermore, the lack of matrix's inversion allows the HRP model to reach more robust results.

The algorithm provides that the clusters in $(y_{N-1,1}, y_{N-1,2})$ are iteratively replaced with their constituents, until only the original elements and no cluster remain. The output of the algorithm is a sorted list of the unclustered items, in this way the order of the clustering is preserved. The results are more intuitive looking at Figure 6 and Figure 7, where the original covariance matrix and the clustered covariance matrix are reported. In the last one, the formed blocks can be interpreted as clusters of assets.

Figure 6 - "Heatmap of original covariance matrix"

Data source: López De Prado (2016, p.22)

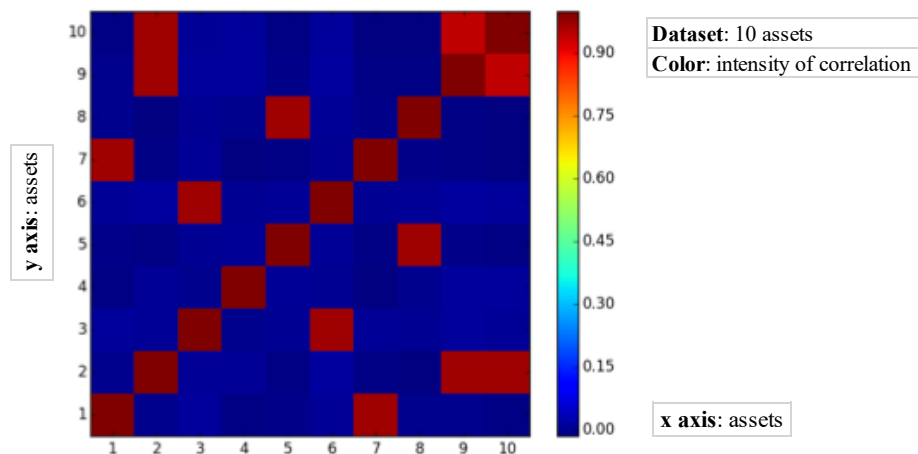
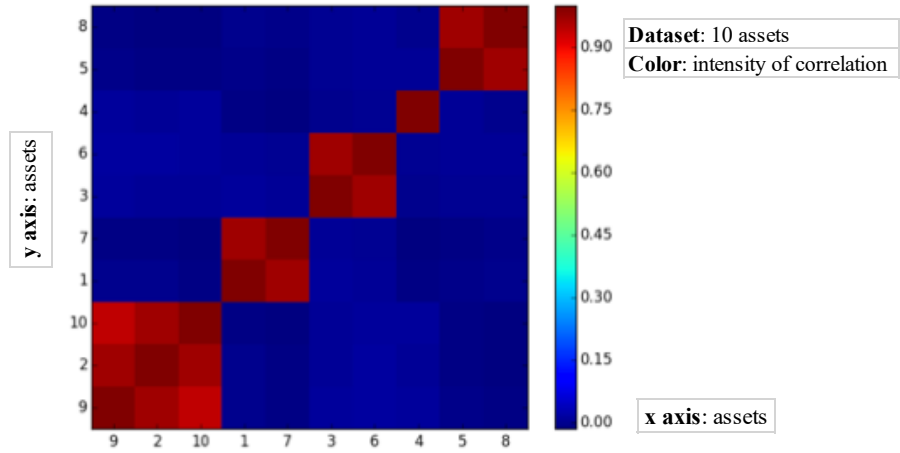


Figure 7 - "Heatmap of clustered covariance matrix"

Data source: López De Prado (2016, p.24)



3.1.3 Recursive Bisection

The third stage for the algorithmic implementation of the HRP model is the recursive bisection, where the allocation of weights is performed. In the previous stage of quasi-diagonalization, the result was a quasi-diagonal covariance matrix. As derived by López De Prado (2016), when the covariance matrix is diagonal, the inverse variance allocation is optimal. With this type of allocation, the weight within a cluster is allocated inversely with respect to the variance of the subsets. To analyze the inverse variance allocation, it is necessary to consider N items and to examine the optimization problem where the objective is to minimize the portfolio's variance, subject to the constraint that the sum of weights must be equal to 1 :

$$\begin{aligned} \min_{\omega} \quad & \omega'V\omega \\ \text{s. t.} \quad & \omega'a = 1_I. \end{aligned}$$

In this case, with the inverse variance allocation, the weight is $\omega = \frac{V^{-1}a}{a'V^{-1}a}$. Considering $a = 1_N$, the solution gives the minimum variance portfolio. When the matrix V is diagonal the resulting ω_n is $\frac{V_{n,n}^{-1}}{\sum_{i=1}^N V_{i,i}^{-1}}$. In the recursive bisection the allocation is done with a top-down approach where the assets are reiteratively divided in bisections. The division is always made

in half so the items to consider are 2. Therefore, $\omega_1 = \frac{\frac{1}{V_{1,1}}}{\frac{1}{V_{1,1}} + \frac{1}{V_{2,2}}} = 1 - \frac{V_{1,1}}{V_{1,1} + V_{2,2}}$.

The algorithm to implement in the recursive bisection stage can be described through six different steps.

Firstly, the list of items, so of assets, in the portfolio is set and it is denoted as $L = \{L_0\}$, with $L_0 = \{n\}_{n=1,\dots,N}$. A vector of unit weights assigned to every item is initialized as $w_n = 1, \forall n = 1, \dots, N$.

Secondly, the algorithm stops if $|L_i| = 1, \forall L_i \in L$.

Thirdly, for each $L_i \in L$ such that $|L_i| > 1$:

- a. split L_i into two subsets, $L_i = L_i^{(1)} \cup L_i^{(2)}$, where $|L_i^{(1)}| = \text{int} \left[\frac{1}{2} |L_i| \right]$ and in this way the order is maintained;
- b. the variance of the partition $L_i^{(j)}$ is computed as the variance of an inverse-variance allocation. In this passage a bottom-up approach is used to identify the variance of continuous subsets. The variance of the subset $L_i^{(j)}, j = 1, 2$ corresponds to $\tilde{V}_i^{(j)} \equiv \tilde{w}_i^{(j)'} V_i^{(j)} \tilde{w}_i^{(j)'}$, using the inverse variance weighting,²⁵
- c. the top-down approach is used to identify the weights to allocate to the adjacent subsets. Once again, this is done applying the inverse variance allocation. The split factor α_i is computed as $1 - \frac{\tilde{V}_i^{(1)}}{\tilde{V}_i^{(1)} + \tilde{V}_i^{(2)}}$, hence $0 \leq \alpha_i \leq 1$;
- d. the computed allocations w_n are repropotioned according to the split factor α_i , for I subset of the bisection ($\forall n \in L_i^{(1)}$)
- e. and the weights w_n are repropotioned by a factor $(1 - \alpha_i)$, for the other subset of the bisection ($\forall n \in L_i^{(2)}$).

Fourthly, the algorithm loops to step two.

This process also avoids short-selling $0 \leq w_i \leq 1, \forall i = 1, \dots, N$ and it allows the allocation of the entire wealth $\sum_{i=1}^N w_i = 1$. The latter constraint is respected because, in allocating the weights, the proportions are iteratively maintained since the weights come from a superior hierarchical level.

²⁵ López De Prado (2016, p.8) specifies that " $V_i^{(j)}$ is the covariance matrix between the constituents of the $L_i^{(j)}$ bisection, and $\tilde{w}_i^{(j)} = \text{diag} [V_i^{(j)}]^{-1} \frac{1}{\text{tr}[\text{diag}[V_i^{(j)}]^{-1}]}$, where $\text{diag}[\cdot]$ and $\text{tr}[\cdot]$ are the diagonal and trace operators."

3.2 Performance in-sample

In order to have concrete evidence of the improvements theoretically expected with the implementation of the Hierarchical Risk Parity model, the author López De Prado (2016) carried out a numerical example. The author generated random data and then he applied and compared the HRP model, the CLA model and the Inverse-Variance Portfolio (IVP). The CLA model is implemented as representative of the quadratic optimizers, while the IVP is considered as representative of the traditional risk parity allocation methods. In particular, for the CLA the minimum variance allocation is performed given that it is the unique portfolio along the efficient frontier which relies only on risk and so it is more comparable with the other risk-parity techniques implemented.²⁶

The inputs data are generated by simulating a matrix X of observations of returns, with a size of 10000×10 . On these data the algorithm previously outlined is implemented, so the dendrogram, the clustered correlation matrix and the weights for each asset are obtained. There is to point out that, in this case, the condition number of the covariance matrix is not so high, this means that the CLA should not deliver too unstable results. As expected, the tree structure groups together the series of returns which are calculated as perturbations of a specific series.

Table 1 - "A comparison of three allocations"

Data source: Reworked table in Excel from López De Prado (2016, p.25)

Portion of wealth in % allocated to each asset

Asset	Type of allocation		
	CLA	HRP	IVP
1	14.44%	7.00%	10.36%
2	19.93%	7.59%	10.28%
3	19.73%	10.84%	10.36%
4	19.87%	19.03%	10.25%
5	18.68%	9.72%	10.31%
6	0.00%	10.19%	9.74%
7	5.86%	6.62%	9.80%
8	1.49%	9.10%	9.65%
9	0.00%	7.12%	9.64%
10	0.00%	12.79%	9.61%

CLA: Critical Line Algorithm
HRP: Hierarchical Risk Parity
IVP: Inverse Variance Portfolio

²⁶ The default constraints of no short selling $0 \leq w_i \leq 1, \forall i = 1, \dots, N$ and full investment $\sum_{i=1}^N w_i = 1$ are implemented.

On one hand, looking at the allocation of weights reported in Table 1, it is possible to see how the CLA places the 92.66% of wealth on top five assets, and 0% on other three specific assets.²⁷ The concentrated allocation of the CLA derives from the fact that the objective of the method is to minimize the portfolio variance. Therefore, the CLA's minimum variance portfolio tends to invest more in the assets that allow reaching a lower level of risk, even dropping a part of the universe of assets. This type of allocation exposes the portfolio to idiosyncratic shocks which affect the single investments.

On the other hand, the IVP allocates the wealth in an equal manner among all assets, disregarding the correlations between them. This type of allocation exposes the portfolio to systemic shocks which affect the entire universe of investments.

The HRP allocates only the 54.18% of the entire wealth on top five assets, so, with respect to the CLA, the effect of a negative shock on those five assets will have a lower impact on the portfolio.

In this way, the model allows to reach a balance between the extreme diversification across the entire universe of investments (as delivered by IVP) and the concentrated allocation (as delivered by CLA's minimum variance portfolio). This allocation makes the portfolio more robust in case of shocks, both idiosyncratic and systemic. Furthermore, even if the CLA's allocation could be justified by its objective, the risk of the portfolio is not so much lower than in the case of implementation of the HRP. The CLA reaches a standard deviation $\sigma_{CLA} = 0.4486$, slightly lower with respect to the one reached by the HRP, $\sigma_{HRP} = 0.4640$. This numerical example represents the in-sample analysis carried out by López De Prado (2016).

²⁷ The allocations on these last three assets would probably have been negative in absence of the non-negativity constraints.

3.3 Performance out-of-sample

The in-sample results may be due to the specific dataset on which the analysis has been carried out. The same portfolios implemented on a different kind of dataset could lead to different results, but there is also the risk that the set of data is defined according to some optimal parameters that allow the HRP to achieve better performance. In this way the model would be based on specific observations instead of a general structure, leading to overfitting bias. To reach realistic results and to avoid eventual overfitting bias, it is useful to check if the in-sample and the out-of-sample analysis achieve the same results. López De Prado (2016) performed a Monte Carlo simulation to compare the allocations out-of-sample of the CLA, the HRP and the IVP and to identify the one that performs best irrespective of circumstances.

As explained by Raychaudhuri (2008), the Monte Carlo simulation provides random sampling, in order to avoid the use of the exact training set exploited for the in-sample analysis. This process is reiterated a predetermined number of times, in order to perform a series of analyses and deliver numerical results. To reproduce the daily returns of 10 assets with a time horizon of 2 years, the author in his experiment generated 520 observations for 10 series of random Gaussian returns. The observations are set to have 0 mean and a standard deviation of 10%. To make the setting more realistic, random shocks, both positive and negative, have been added to the series to reflect the frequent movements in prices, together with a random correlation scheme which reflects the correlation between returns from different assets. The portfolios are computed taking into account the more recent 260 observations, corresponding to one year of observations. The allocations of the portfolios are computed according to a rebalancing period of one month, corresponding to 22 daily observations. To compare the three different methods, the performances of returns for each model are computed and compared. To avoid the biases linked to the dataset identification, the process is reiterated 10000 times.

As expected, given the initial assumptions, the returns of the portfolios have zero mean, while the differences lie in the variances. The CLA's minimum variance portfolio, contrary to its objective of minimizing the risk and so the variance, delivers $\sigma_{CLA}^2 = 0.1157$ which is 72.47% higher than the HRP's portfolio variance $\sigma_{HRP}^2 = 0.0671$. Therefore, the Sharpe ratio of the HRP turns out to be 31.3% greater than the one reached with the CLA. The variance of the CLA is also higher than the variance of the IVP which is $\sigma_{IVP}^2 = 0.0928$. Furthermore, the IVP's variance is 38.24% higher than the HRP portfolio's variance. In conclusion, the HRP model

allows to achieve a lower variance, so a lower risk out-of-sample. This result is particularly significant for the investors that make extensive use of leverage, such as those who implement risk parity techniques.

The author of the simulation also analyzed how the weights evolve across time according to each model, producing three different time series of weights' allocations. What is relevant is the reaction of the three different portfolios when facing shocks that increase the volatility of one or more assets. From the time series, it emerges that the IVP every time that faces a shock, related to a specific investment or common to all investments, reacts by decreasing the portion of wealth allocated to the affected assets. Consequently, that portion of wealth is shifted on all the other unaffected assets. In this way, the weights assigned to the unaffected assets automatically increase over time, without any consideration of the underlying correlation scheme, and this implies a loss of diversification. The correlation among those investments should be considered in order to avoid large negative impacts coming from systemic shocks and to avoid excessive rebalancing. As regards the CLA, the allocation of weights varies in a very irregular way when reacting to shocks, with the consequent increase of the rebalancing costs. The HRP, with respect to the IVP, considers the correlations. In this way, when an idiosyncratic shock occurs, the model decreases the allocation to the affected investment and increases the weight of an unaffected correlated one. In the case of a shock that involves several assets, the HRP reduces the investment in those assets, as the IVP, but it augments the investment only in the uncorrelated assets that present smaller variance. So, it achieves a more thoughtful and meaningful allocation that already offers good protection in distress conditions and avoids extreme rebalancing.

The results seem to be in accordance with the intuitive reasoning drawn from the conclusions of the in-sample analysis; the concentrated CLA portfolio is highly exposed to idiosyncratic shocks, while the IVP is highly exposed to systemic shocks, given that it focuses exclusively on the variances of the single assets without incorporating the information coming from the correlations. Again, the HRP performs better in case of both idiosyncratic and systemic shocks, its allocation lies between the concentration of the CLA and the diversification of the IVP.

According to López De Prado (2016), by changing the parameters of the simulation it is possible to observe that the better performance of the HRP model out-of-sample is even more glaring in certain situations. For example, when additional assets are included in the dataset, or when a greater number of random shocks or a more accentuated cross-sectional correlation is

introduced, or, lastly, when the costs arising from the portfolio's rebalancing process are assessed.

The HRP algorithm is flexible because it allows to replace the correlation matrix, the distance matrix, the clustering algorithm, the linkage criteria and so on. In [Appendix B](#), some modifications proposed or performed by different authors are reported.

Summing up, the Hierarchical Risk Parity model brings together different methods for overcoming the instability issues of the quadratic optimizers. By applying the steps of tree clustering, quasi-diagonalization and recursive bisection, a numerical example and a Monte Carlo simulation have been carried out on simulated data by the author of the model. The results show how, with the HRP, is possible to achieve greater stability, greater diversification and lower volatility²⁸ with respect to quadratic optimizers and traditional risk parity allocations. Thanks to these characteristics the model delivers greater protection from idiosyncratic and systemic shocks, therefore it could be considered a really valid alternative for portfolio allocation problems.

²⁸ Lower volatility is reached by the HRP in the out-of-sample analysis.

4. EMPIRICAL APPLICATION

The theoretical discussion carried out so far has pointed out the outperformance of the HRP model out-of-sample, with respect to the CLA minimum variance portfolio and to the IVP. It could be interesting to see how the HRP behaves in practice when it is applied to real data instead of generated data. In this chapter, an empirical application of the HRP model is done on a portfolio composed of equity Exchange Traded Funds (ETFs). A backtest is performed by using the information derived from the real prices of these financial instruments for a time horizon of 11 years, from 2011 to 2021. In order to evaluate the performance of the HRP, a comparison with the Markowitz's mean-variance framework is made. In particular, I consider the Global Minimum Variance (GMV) portfolio, which is the portfolio with the lowest variance among those efficient. To this portfolio I applied two different sets of constraints that are described in [subsection 4.2](#). At first, the different allocations are compared along the entire sample, then during two different time windows and, lastly, the in-sample and out-of-sample performances are analyzed.

In the next subsections, I present the selected dataset and the reasons why the analysis is performed on a portfolio composed exclusively of equity ETFs ([subsection 4.1](#)). Then, I introduce the methodology implemented to compare the HRP and the Markowitz's framework ([subsection 4.2](#)). Lastly, I present and comment the results providing some concluding remarks ([subsection 4.3](#)).

4.1 Dataset

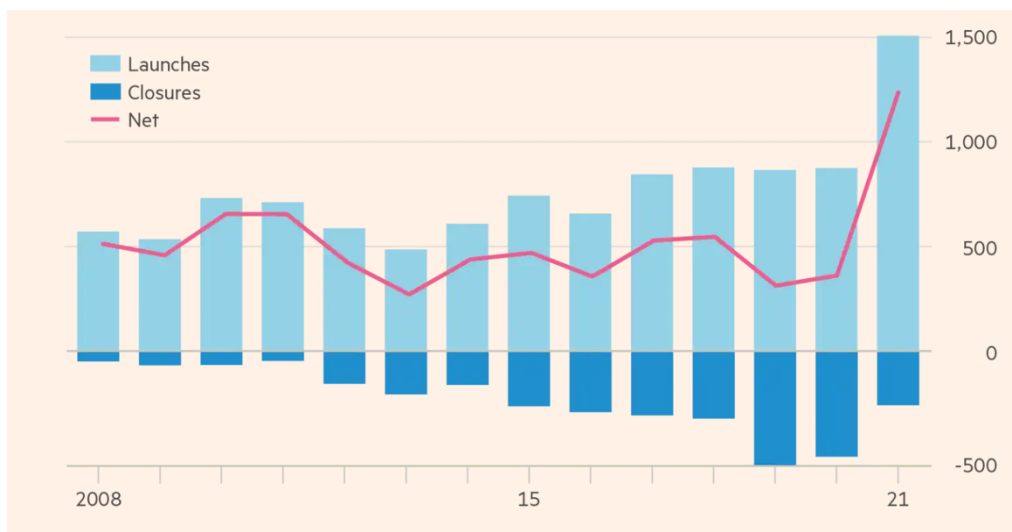
In the last year, the financial instruments called ETFs have become very popular among financial investors. Since their approach on the market, these instruments have seen an increasing success, linked to the increase in the volumes of trade and to the increased size of the managed masses. At the end of July 2021, according to ETFGI²⁹, the size of the global ETF

²⁹ ETFGI is an independent research and consulting firm, which analyzes a variety of financial products on a global scale.

market corresponded to 9.464 trillion U.S. dollars.³⁰ In addition, the number of ETFs available on the markets has exponentially increased.

The explosions of the ETFs in 2021 can be clearly seen in Figure 8, where the number of launches of newly formed ETFs far exceeded the number of ETFs that have been closed, leading to a sharp increase in the net amount. As reported in the article of the Financial Times by Johnson (2022), this high number of launches can be justified, in Europe, by the increasing importance recognized to Environmental, Social and Governance (ESG) standards, which led to the foundation of new sustainable ETFs and, in the United States, by the increased number of active ETFs opened.

Figure 8 – “ETF launches and closures globally, by year” Data source: Johnson (2022)- Financial Times



The reasons for this rush to ETFs come from their accessibility and from the advantages implied in their structure. The principal advantages of ETFs are the simplicity, the transparency, the flexibility, the cost-effectiveness, and the reduction of the issuer’s risk. The detailed descriptions of these benefits of investing in ETFs can be found in the next [subsection 4.1.1](#), together with the definition of the structure of these instruments.

³⁰ These data are available at the website <https://etfgi.com/news/press-releases/2021/08/etfgi-reports-assets-global-etfs-industry-reach-record-us946-trillion>.

4.1.1 Exchange Traded Fund (ETF)

The European Markets in Financial Instruments Directive (MiFID II, Directive 2014/65/EU) defines ETF as “ a fund of which at least one unit or share class is traded throughout the day on at least one trading venue and with at least one market maker which takes action to ensure that the price of its units or shares on the trading venue does not vary significantly from its net asset value and, where applicable, from its indicative net asset value”.

The main goal of the ETFs is to track and to reproduce the performance, for example, of a particular index, a particular sector or a specific investment strategy. There exist different types of ETFs, based on the objective they pursue. An ETF could focus on a specific asset class, a specific thematic (such as artificial intelligence, gender equality, future mobility, etc.), a specific geographic area and so on. The most substantial classification splits the ETFs according to the kind of tracking that they perform, which can be “passive” or “active”. The following implementation is done on passive ETFs, also called Indexed-Based ETFs. They represent the largest portion of ETFs traded in the market and their objective is to replicate the performance of a specific index, especially of equities indexes.

These instruments have several advantages. They are simple because they allow the investor to directly enter in the target market (equity, bonds, commodities, etc.) or to directly implement the target strategy (for example short and leverage strategies). Furthermore, they can be easily purchased or sold through the bank or the broker. There is proper transparency; given that usually notorious indexes are tracked, the components of the reference index can be easily identified by the investor through the publicly available information.³¹ They are flexible because it is possible to invest starting with only one share so even by investing a small amount of money and the investor can define the time horizon of the investment according to his own objective. They reach cost-effectiveness thanks to the passive management and so to the lower amount of operating costs. Lastly, ETFs are funds whose assets are defined by law as exclusive property of the holders of the shares; in case of insolvency of the company that manages the fund, the assets of the ETF would not be affected, implying a reduction of the issuer’s risk.

³¹ In addition, the Net Asset Value (NAV) is published daily. The NAV corresponds to “the value of the ETF’s assets minus its liabilities divided by the number of shares outstanding”, as reported by the SEC (2012, p.1). The liabilities of the ETF are, for example, the commissions paid to the society in charge of managing the fund.

4.1.2 The selected ETFs

Given the rising importance of ETFs in financial markets and the advantages of the Indexed-Based ETFs, the dataset for the empirical application is exclusively formed by ETFs which passively track the performance of some indexes. More specifically, the tracked indexes come from the universe of investments in equities, given that equity ETFs are the most popular kind of ETFs. In this way, the implemented allocations will represent a 100% equity strategy. Assuming that the investor in question is a young person who wants to implement a long-term investment strategy, the 100% equity strategy seems to be a good choice. Siegel (2014) studied that the most secure investment with a long-term horizon is a diversified portfolio of equities; this is due to the fact that when the stocks are invested for a long period of time, the probability of outperforming fixed-income instruments increases a lot. Considering an investment period of 30 years, stocks beat bonds almost 100% of the time. This is the reason why only equity ETFs have been included in the dataset.

In order to have a universe of assets which can be clustered according to an intuitive criterion, the dataset is composed of ETFs that track the major equities indexes in different regions of the world. In this sense could be interesting to check if assets tracking geographic areas from the same continent are grouped together. In fact, the assets could be grouped according to different criteria: asset class, geographical area, sector or size. The selected ETFs are presented in Table 2.

The daily prices of equity ETFs are downloaded from the software EIKON v-4, using Thomson Reuters Datastream. All the prices have been retrieved in Euro currency, so all the computations are carried out using the Euro as currency.

Table 2 - Selected ETFs

Data source: Elaboration in Microsoft Excel³²

Ticker	Name	Benchmark	Style Benchmark	Geographic Area
IWRD	ISHARES MSCI WORLD (AMS) UCITS ETF USD (DIST)	MSCI WORLD INDEX	Developed	World
C40	AMUNDI CAC 40 UCITS ETF DR - EUR (A)	CAC40	Developed	Europe
CG1M	AMUNDI ETF DAX UCITS ETF DR C/D	DAX	Developed	Europe
FMI	AMUNDI ETF FTSE MIB	FTSE MIB	Developed	Europe
ISF	ISHARES CORE FTSE (MIL) 100 UCITS ETF GBP	FTSE 100	Developed	Europe
A500	AMUNDI ETF S&P 500 (MIL)	S&P500 TRN USD	Developed	North America
FLXB	FRANKLIN FTSE BRAZIL UCITS ETF	FTSE BRAZIL 30/18 CAPPED NET TAX INDEX	Emerging	Central/South America
CSCA	ISHARES MSCI CANADA(MIL) UCITS ETF USD (ACC)	MSCI CANADA TRN	Developed	North America
ISVE	ISHARES MSCI AUS. (BER)	MSCI AUSTRALIA TRN	Developed	Pacific
ISVW	ISHARES MSCI SA. (BER) IDX.FD.	MSCI SOUTH AFRICA 20/35 INDEX	Emerging	Africa/Middle-East
HMCE	HSBC MSCI CHINA (MIL) UCITS ETF USD	MSCI CHINA INDEX	Emerging	Asia
JPNA	UBS LUX FD.SLTN. (MIL) MSCI JAP.UCITS ETF JPY	MSCI JAPAN	Developed	Pacific
XFVT	XTRACKERS FTSE VTM.(MIL) SWAP UCITS ETF 1C	FTSE VIETNAM TR	Emerging	Asia
HMIU	HSBC MSCI INDONESIA(MIL) UCITS ETF USD	MSCI INDONESIA TRN	Emerging	Asia

From the different tests performed on the HRP model in the previous literature, it emerges that at least 10 years of observations are needed to have an adequate sample. The timespan of the downloaded prices goes from 2011 to 2021. There is also to point out that the dataset includes an ETF that tracks the MSCI World Index³³, which is focused on the world area and it includes equities only from developed countries. It will be used as a benchmark in comparison with the created portfolios which, instead, will also include data from emerging countries (such as South Africa and Vietnam), in order to cover the African continent and to be more diversified. The indexes referred to Brazil (ticker “FLXB”), China (ticker “HMCE”) and Indonesia (ticker “HMIU”) have been discarded from the sample given the lack of data from 2011 to 2019. Regarding the developed countries, I have selected for each continent those included in the world index with the best ESG scores and with enough daily observations available.

In this way, the final dataset is composed of a dataframe of 10 columns, each one representing one asset, and 2872³⁴ rows, each one displaying the daily price of each asset for 11 years.

³² The program used is Microsoft® Excel for Mac version 16.57.

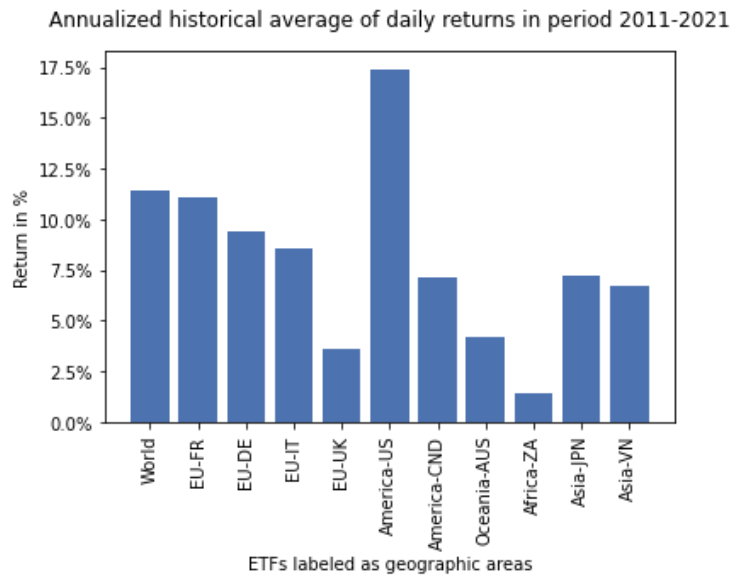
³³ As reported in the website <<https://www.msci.com/documents/10199/149ed7bc-316e-4b4c-8ea4-43feb5bd6523>>, “the MSCI World Index captures large and mid cap representation across 23 Developed Markets (MD) countries... DM countries include: Australia, Austria, Belgium, Canada, Denmark, Finland, France, Germany, Hong Kong, Ireland, Israel, Italy, Japan, Netherlands, New Zealand, Norway, Portugal, Singapore, Spain, Sweden, Switzerland, the UK and the US.”

³⁴ The average number of trading days in a year is assumed to be 252.

4.1.3 Descriptive statistics of the ETFs

Firstly, the fundamental descriptive statistics for the different ETFs are computed. From the graphs, it is possible to spot the differences in terms of historical average returns, historical average volatilities and correlations. The code snippets for the computation can be found in [Appendix C.2](#).

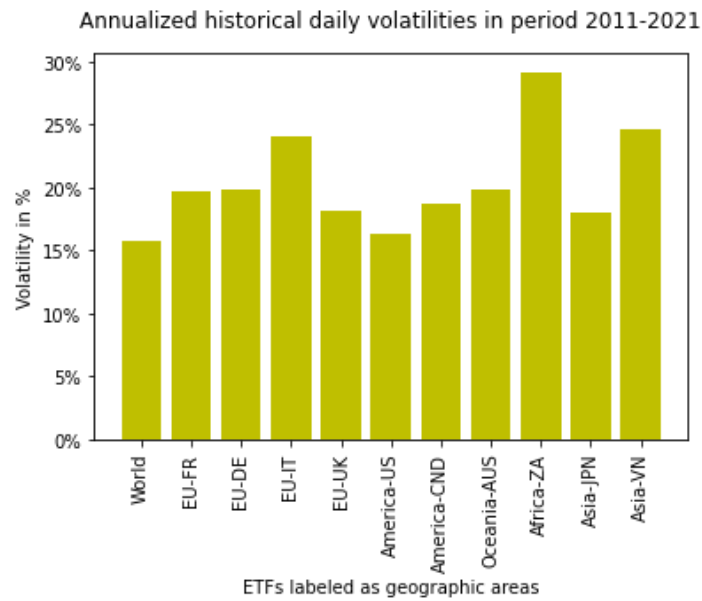
Figure 9 - Historical Returns ETFs Data source: Elaboration in Python



In Figure 9 there is the bar plot representing the historical daily returns, which are computed as annualized means of returns. After the calculation of the returns, the mean $\mathbf{r} = \frac{1}{N} \sum_{j=1}^T r_t$ is computed and then annualized.³⁵ At first sight, the ETFs which present the highest expected return are the ETFs tracking the U.S., the world area and France; while the lowest expected returns are delivered by the ETFs tracking South Africa, U.K. and Australia.

³⁵ The annualization is done by taking $(\mathbf{r} + 1)^{252} - 1$.

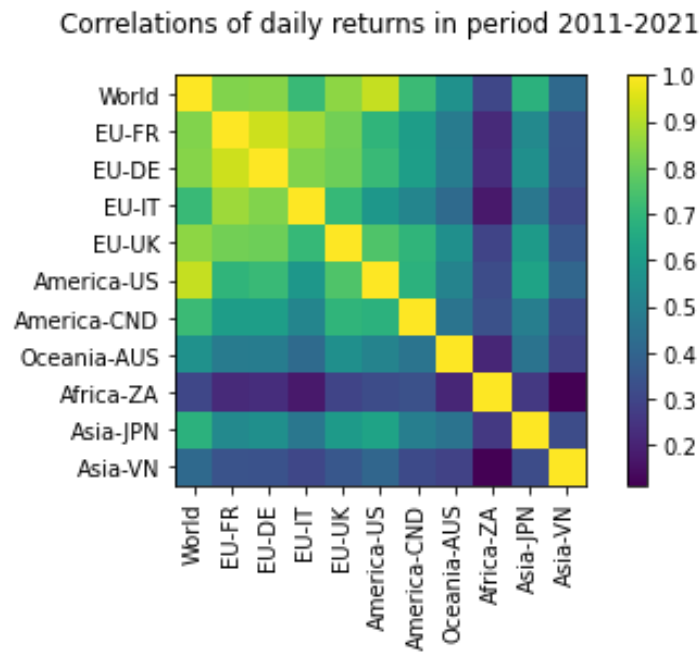
Figure 10 - Historical Volatilities ETFs Data Source: Elaboration in Python



The annualized historical daily volatilities of returns are represented through a bar plot in Figure 10. The covariance matrix of returns is computed as $\Sigma = \frac{1}{N} \sum_{j=1}^T (r_t - \mathbf{r})(r_t - \mathbf{r})'$; then the application of the square root delivers the volatility.³⁶ The countries which present the most volatiles returns are South Africa, Vietnam and Italy. Indeed, two out of three are emergent countries. The geographic areas with more stable returns seem to be the World area, the U.S. and Japan.

³⁶ In the case of standard deviation, the annualization is done just by multiplying it by 252, the number of trading days in one year.

Figure 11 - Correlations of the ETFs Data source: Elaboration in Python



To plot the surface of correlations, as reported in Figure 11, the correlations of each asset with one another are calculated.³⁷ Looking at the surface, it is possible to see how all the European countries are highly correlated and how those countries, together with the U.S., show high correlations with the ETF World. This means that Europe and the U.S. play an important role in determining the evolution of the ETF World.

³⁷ The correlations are computed according to the Pearson's method.

4.2 Methodology

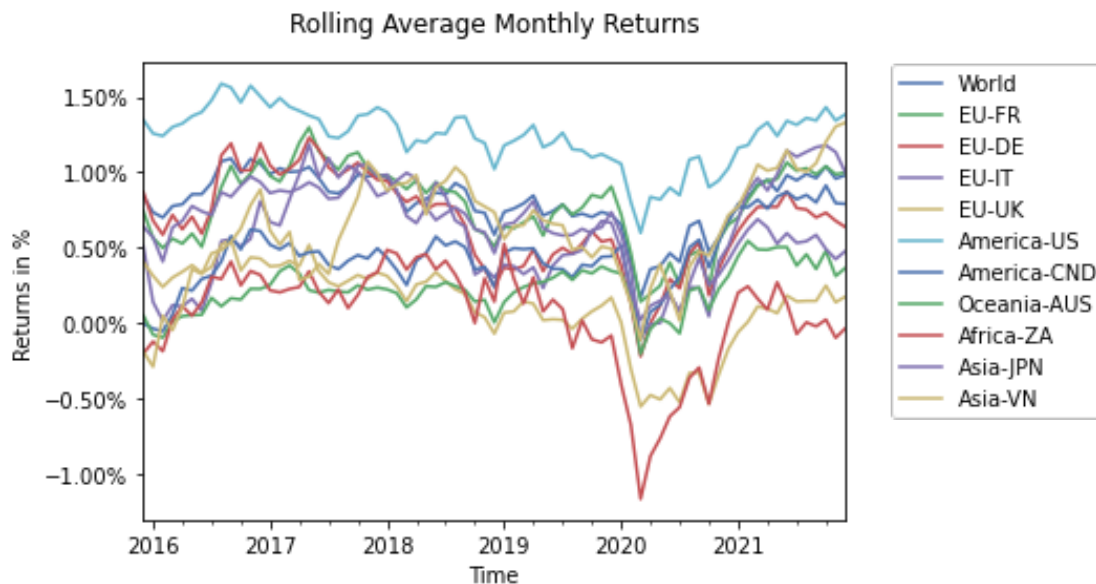
The focus of the empirical analysis is the application of the HRP model to real data. In order to analyze the performances of the model, a comparison with other methods of portfolio allocation is carried out. The HRP, which is a risk parity portfolio, is compared to the GMV portfolio, which is the only portfolio based exclusively on risk along the efficient frontier.³⁸ To the GMV, two different sets of constraints are applied to produce two different portfolios. The first portfolio is constructed considering the positivity constraints, meaning that short-selling is not allowed because the lower bound for weights is set to zero. The second portfolio, which for simplicity will be labeled as “GMV constrained”, is constructed still considering the positivity constraint and that constraint is improved to a minimum of 3% with the objective to achieve higher diversification. Moreover, other lower and upper bounds are imposed on group constraints. On one hand, the two emerging countries, Vietnam and South Africa, are the areas which present low returns and high volatilities, so an upper bound is imposed: up to 25% of total wealth can be invested in this group of countries. On the other hand, the U.S. is the country that allows reaching the most profitable and the least volatile returns, so a lower bound of 40% is imposed.

The inputs used for Markowitz’s model are the sample moments; in this case the annualized mean of daily historical returns and the annualized sample covariance matrix of daily asset returns are used. The inputs of the HRP are only the daily returns. The daily returns r_t are computed as the change in the price from day $t-1$ to day t , with respect to the price at time $t-1$, so as $\frac{p_t - p_{t-1}}{p_{t-1}}$, where p_t corresponds to the price at time t . For the computation of the allocations, the first asset, which corresponds to the ETFs tracking the World Index, is discarded from the returns. The portfolios will be calculated only considering the remaining assets because the ETFs World is used as a benchmark for comparison purposes.

³⁸ In this particular case, where the dataset is not so large and specific constraints are implemented, the results delivered by the CLA minimum variance correspond to the results delivered by the GMV portfolio. The equality of the two methods has been proven by implementing the CLA minimum variance and the GMV with the functions offered by the PyPortfolioOpt library, as described in the next subsections. The results that they deliver are the same in terms of allocations, expected return and volatility.

Three different analyses are carried out to evaluate the performance in terms of expected return, volatility and Sharpe ratio.³⁹ The first analysis is carried out considering the entire sample from 2011 to 2021. The second analysis provides a comparison to see how the ETFs' portfolios performed during two different subperiods, one called "Pre-Covid" which goes from January 2011 to the 24th of February 2020 and one called "Post-Covid" which goes from the 24th of February 2020 to December 2021. These two subperiods have been identified by looking at the plot of monthly rolling returns in Figure 12. The plot allows individuating a particular moment where a sharp decrease took place and negative returns were reached; that moment corresponds to the start of the Covid.

Figure 12 - Monthly rolling returns Data source: Elaboration in Python



The third analysis is a comparison of the performance in-sample versus out-of-sample. The in-sample performance corresponds to the one computed on the entire sample, while the out-of-sample is evaluated implementing the "rolling sample" approach. This means that the inputs moments are computed by using a rolling estimation window w over the entire dataset. The window length is of 1260 daily observations, equivalent to 5 years. The weights are rebalanced every 21 trading days, which corresponds to a period of one month. Each loop for the computation of the portfolios includes different steps and it is applied for each method:

1. Starting at $t=w$, the sample mean and covariance matrix are estimated over the previous 1260 days, given that $w=1260$.

³⁹ The Sharpe ratio is computed considering a risk-free rate of 0.02.

2. The next step corresponds to using the previously computed moments to solve the optimization problem and so to obtain the portfolio weights for the period $t+1$; this step is repeated every 21 days.
3. The last step involves the computation of the expected return and expected volatility of the portfolios from $t+1$ to $t+21$.

“Rolling the window” means to add the returns of the next 21 days (1 month) and drop the earliest returns each time. Considering the rebalancing period of 21 days, in the second iteration, the loop will take the estimation window from $t=22$ to $t=1281$. The portfolio weights will be computed for $t=1282$ and maintained for the following 21 days. The loop stops when the end of the data T is reached. In this way, the rolling sample approach results in a time series of $T-w$ daily out-of-sample portfolio returns. The Sharpe ratio out-of-sample is calculated taking the mean of the series of portfolio returns and the volatility as the standard deviation of those returns.

The application is carried out using the version 3.8.8 of the programming language Python. In particular, the interfaces Jupyter (6.4.5 version) and Spyder (4.2.5 version) have been used. For the implementation of the different portfolios, the PyPortfolioOpt library (version 1.5.1) developed by Martin (2021) has been used.⁴⁰ The code snippet for the empirical application is available in [Appendix C.2](#). The author López De Prado (2016) elaborated the source code for the implementation of the HRP algorithm in Python. In [Appendix C.1](#), the source code applied to the current dataset is reported. It has been verified, with a concrete application, that the results delivered applying the source code for the HRP model correspond exactly to the results delivered directly applying the commands for the allocations of weights from the library PyPortfolioOpt. For this reason, the commands of the library are directly implemented. In order to implement the code, the user just needs to modify the path “/Users/ilariapadovan/Desktop/Tesi_PythonCode/Dataset.csv”, to upload a different dataset of daily prices in csv format.

⁴⁰ The library is available at the website <<https://pyportfolioopt.readthedocs.io/en/latest/index.html>>.

4.3 Results

The described methodology is applied to the selected dataset of equity ETFs and the results obtained allow to evaluate the performances of the three different kinds of portfolios: the Global Minimum Variance (GMV), the constrained Global Minimum Variance (C-GMV or GMV-C) and the Hierarchical Risk Parity (HRP).

4.3.1 Entire Sample

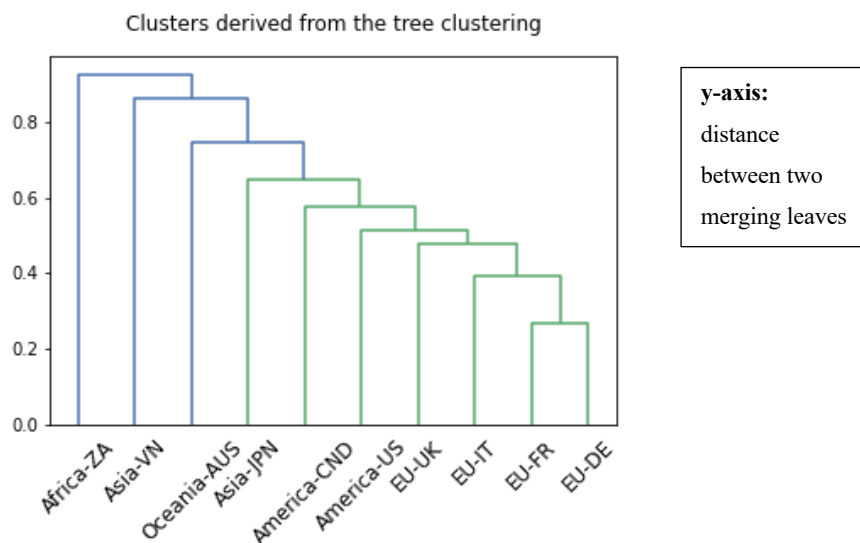
In this first analysis, the different portfolio methods are applied on the entire sample from 2011 to 2021.

First of all, the HRP is analyzed. The first step of the HRP provides the division in clusters of the assets. Looking at the dendrogram in Figure 13 and taking a top-down perspective, it is possible to see that the first two bigger clusters split the ETF tracking South Africa from the rest of the assets. This is probably due to the fact that the South-Africa area presents the lowest return of 14.26% and the highest volatility of 29.29% and these values are very different from the returns and volatilities of the remaining countries. Furthermore, it is possible to say that it divides the unique country representing the African continent, from the other geographical areas. The second partition separates Vietnam from the rest of the countries; a reasonable interpretation of these first two divisions lies in the fact that South Africa and Vietnam are the only emerging countries in the dataset, while all the others are developed countries.

The clusters provided by the algorithm are quite nested, however, they seem to present a geographic meaning. The closest countries are placed in consecutive clusters: Vietnam, Australia and Japan, then Canada and U.S, and at the end U.K., Italy, France and Germany.

Figure 13 - Dendrogram of the ETFs dataset

Data source: Elaboration in Python



The second step of the HRP algorithm provides the quasi-diagonalization of the correlation matrix. From a comparison of the original correlation matrix in Figure 15 and the clustered correlation matrix in Figure 14, it is possible to see that the reorganization of the matrix does not bring big changes because the ETFs are already downloaded in a sort of geographic order. However, looking at the colors in the ordered matrix, it is possible to see how the higher correlations lie along the diagonal. There are no evident color blocks given the nested structure of the dendrogram. Furthermore, in the clustered correlation matrix the ETF tracking the world is excluded because that asset does not take part in the allocation process. Clustering the correlation matrix allows achieving a lower condition number. The condition number for the original correlation matrix corresponds to 220, while the one for the clustered correlation matrix corresponds to 92. A lower condition number means that the matrix is less ill-conditioned, so less prone to be strongly affected by changes in the inputs data.

Figure 15- Surface original correlation matrix

Data source: Elaboration in Python

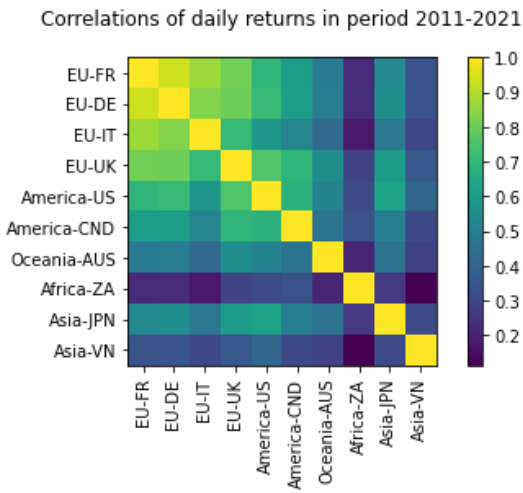
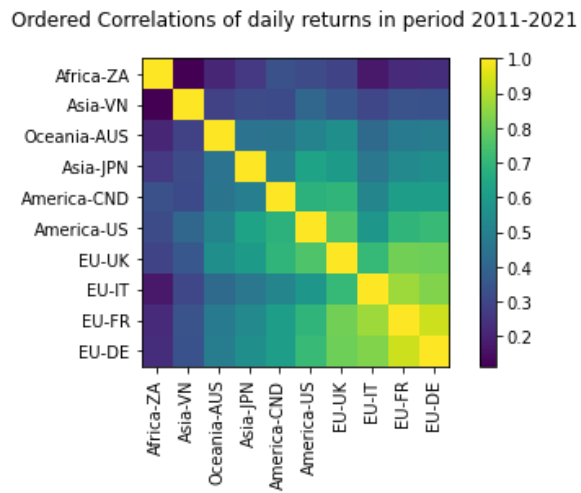


Figure 14 - Surface ordered correlation matrix

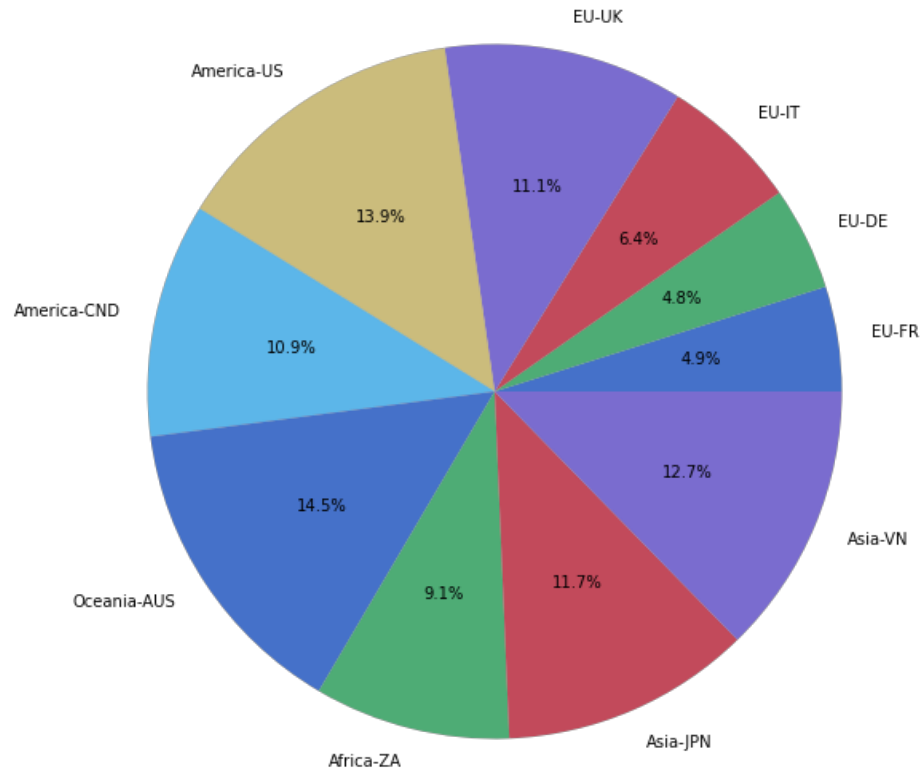
Data source: Elaboration in Python



Thanks to the hierarchical clustering, the quasi-diagonalization of the matrix and the recursive bisection, the weights for the HRP are computed. Looking at the pie chart in Figure 16, it is evident how the HRP allows to reach a really diversified allocation. There are no assets that are discarded in the allocation, but on each individual ETF a portion of wealth is invested. Furthermore, there are no ETFs with particularly dominant weights that prevail over the others. It is also interesting to note how the allocation is geographically diversified. The 24.8% of the total wealth is allocated to U.S and Canada, representing the American continent, the 27.2% is allocated to the European countries, the 24.4% to Asian countries, the 14.5% to Oceania and the 9.1% to Africa. Probably lower weights are assigned to Oceania and Africa, given their returns and volatilities. The single countries (Australia and South Africa) of these areas are a bit overweighted with respect to the single European countries because they are representing an entire continent in the dataset.

Figure 16 – Weights of Hierarchical Risk Parity (2011-2021)

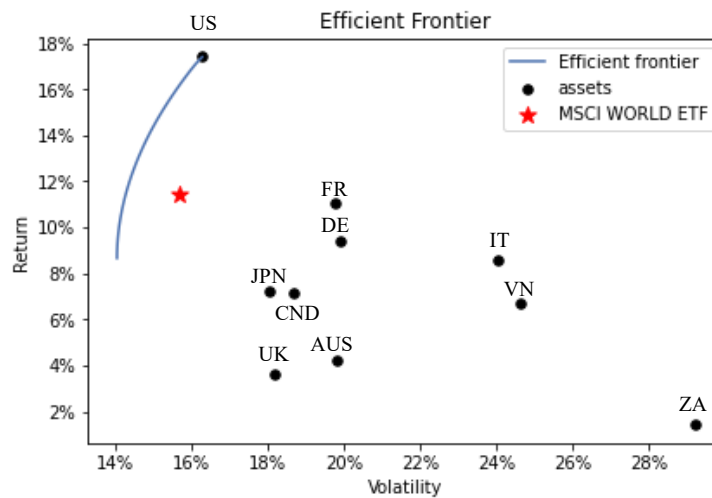
Data source: Elaboration in Python



Moving towards the Markowitz's framework, it is possible to analyze the efficient frontier. Looking at it, displayed in Figure 17, it is immediately observable how the diversification principle works: combinations of assets allow to reach better risk-return levels than the ones achieved by investing in only one asset. On one hand, the U.S. ETF has the highest return and the lowest volatility, indeed it is included at the highest point of the efficient frontier in the upper-left corner of the graph. On the other hand, as expected, the South-Africa ETF is placed in the lower-right corner. Moreover, the World ETF, without considering the U.S., is the one closest to the efficient frontier, indeed it is the most diversified index because it includes all the other assets from all over the world.

Figure 17 - Efficient Frontier from Markowitz's framework

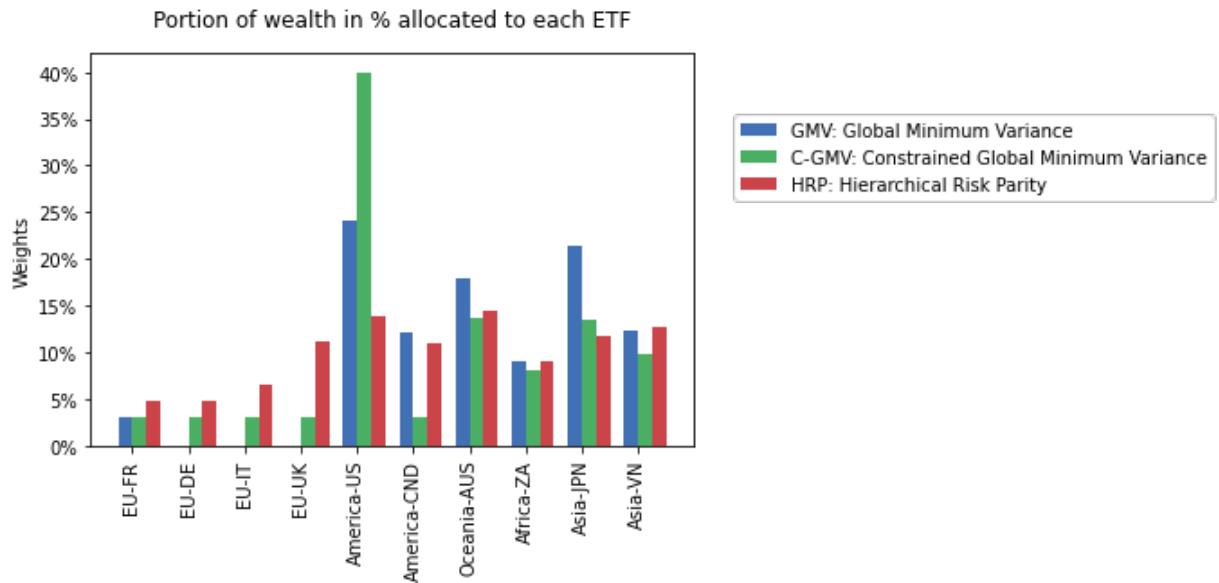
Data source: Elaboration in Python



The weights for the GMV and the GMV constrained (C-GMV) portfolios are derived from the efficient frontier. In Figure 18 the weights of the different portfolios are compared through a bar plot. The first observation to point out is that the HRP delivers the most diversified allocation, while the GMV portfolio presents a concentrated distribution by allocating zero weight to Germany, Italy and U.K. The GMV constrained allocates the 40% of the wealth to the U.S. because of the lower bound imposed on the country. However, it is more diversified with respect to the baseline GMV because the upper bound on the emerging countries and the general lower bound of 3% force it to spread the allocation.

Figure 18- Comparison of weights' allocation in-sample

Data source: Elaboration in Python



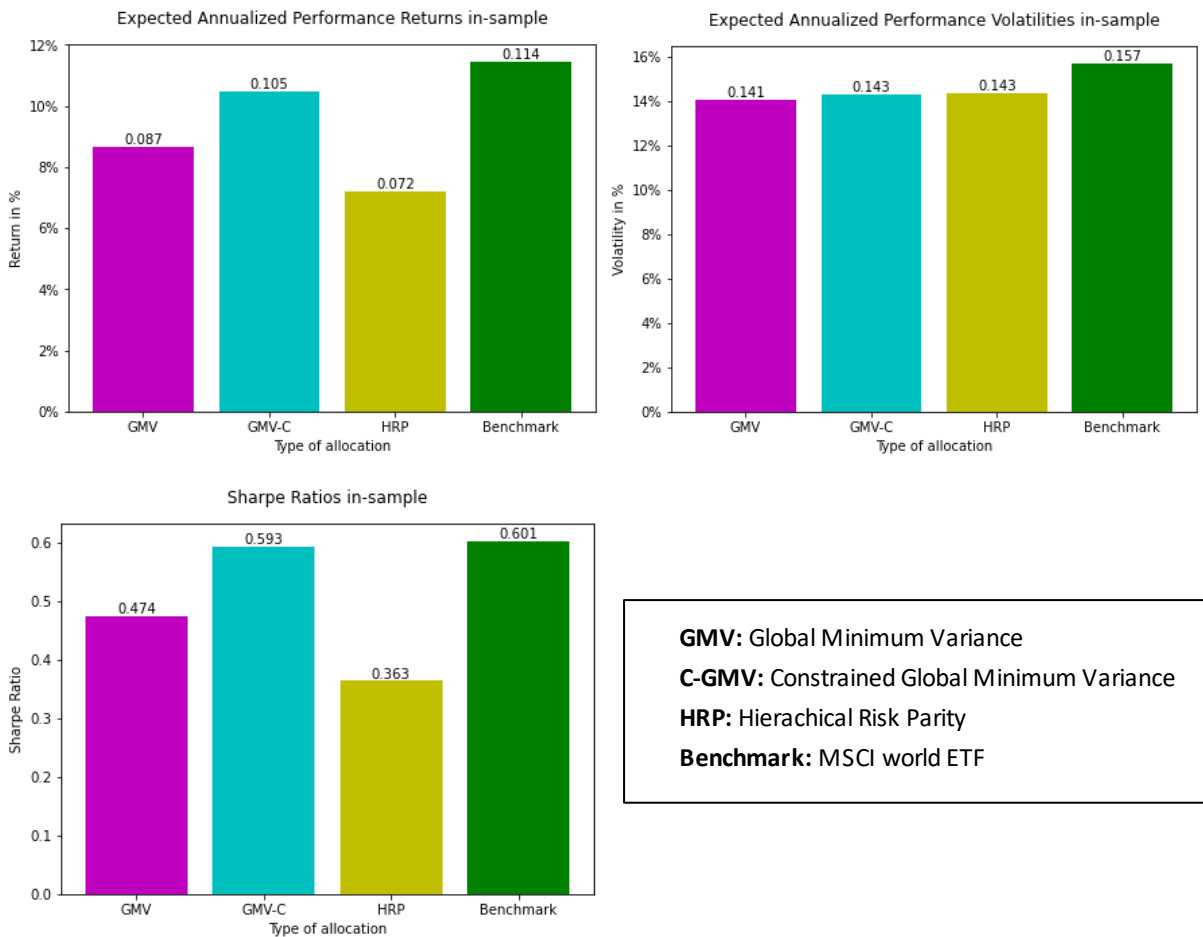
Then the performances of the portfolios and of the benchmark are compared. The expected return, volatility and Sharpe ratio are reported in Figure 19.

The expected performance return of the portfolio is computed by multiplying the expected return of each asset by the weight allocated to that asset. Given that it is a multiplication between two vectors, it corresponds to a sum, which will deliver only one number. The benchmark ETF World presents the highest annual expected return of 11.4%, followed by the GMV constrained and the GMV with 8.7%, while the HRP has the lower expected return of 7.2%. This can be attributed to the fact that the HRP is more diversified; the weights show how it invests less on the most profitable country (U.S.) and it invests more on other countries with lower returns.

The annual volatilities delivered by the different models are very similar. This performance volatility is computed by taking the square root of the variance of the portfolio and annualizing it. The variance of the portfolio is calculated according to the formula $w^T \Sigma w$. It provides the multiplication of the transposed vector of weights with the covariance matrix and, again, with the weight allocated to the assets. Given that it is multiplication between two vectors, it corresponds to a sum, which will deliver only one number. The volatility for the GMV is 14.1%, for the GMV constrained is 14.3% and it is equal to the one delivered by the HRP. The benchmark, instead, delivers a volatility of 15.7%, so a little bit higher than the ones delivered by the portfolios.

Consequently, the Sharpe ratio of the HRP is the lowest one and is equal to 0.36, while the GMV, the GMV constrained and the benchmark present a Sharpe ratio of respectively 0.47, 0.59 and 0.6. According to the Sharpe ratio performance measure, considering the entire sample, the best investment seems to be the World ETF.

Figure 19 - Performance comparison in-sample Data source: Elaboration in Python

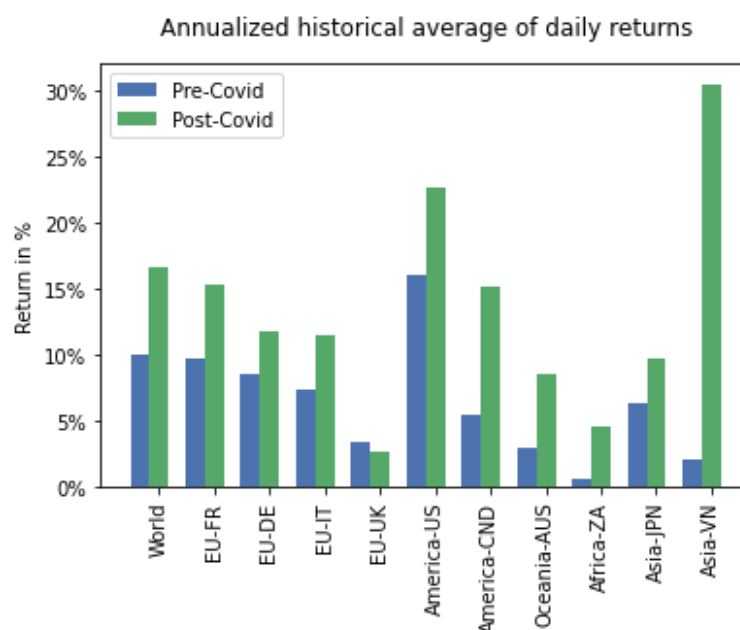


4.3.2 Pre-Covid and Post-Covid

In the second set-up the same analyses are conducted on two different subperiods: “Pre-Covid” and “Post-Covid”.

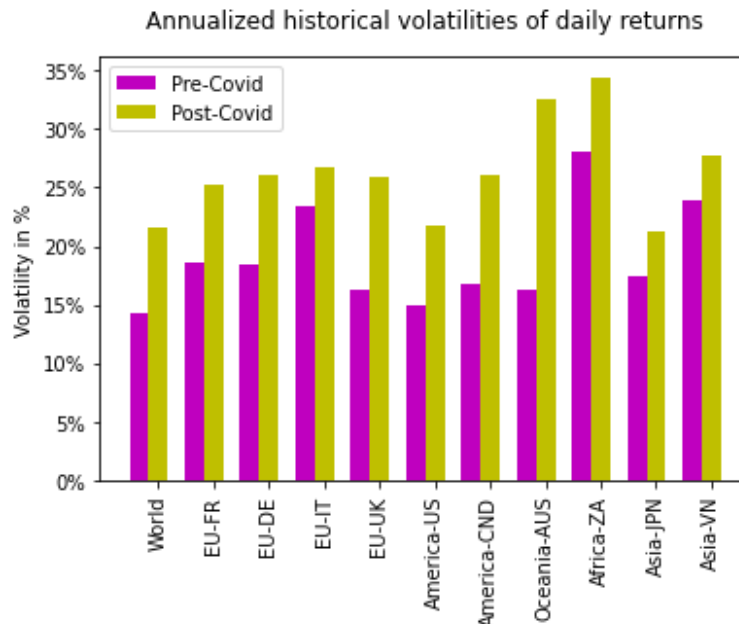
The returns in the two subperiods are plotted in Figure 20. In general, the returns increased after the boom of the Covid. Looking at the movements in prices, it is possible to see the start of decline on the 24th of February 2020, when the Covid spread consistently out of China; then the prices recovered around January 2021. The returns follow the same trend, but during 2021 the prices increased a lot, far exceeding the levels previous to the drop. This could be the reason of the higher return in the “Post-Covid” period. The fact is that the indexes tracked by the ETFs are composed of strong national companies which mostly cover the financial and health sectors and, during the pandemic, these sectors became ever more valued. Furthermore, the financial markets have been highly supported by the policies implemented by the ECB and by the FED. The highest increases affected Vietnam and Canada, probably for policies implemented in response to the Covid. Instead, the U.K. ETF experienced a slight decrease in the returns, probably as a consequence of the Brexit, which has been effective from the 1st of February 2020.

Figure 20 - Comparison of returns Pre-Covid and Post-Covid Data source: Elaboration in Python



Looking at Figure 21, it is possible to see that, after the boom of the Covid, the volatilities increased for all the ETFs. This is reasonable given that the global pandemic brought a lot of uncertainty in the entire world.

Figure 21 - Comparison of volatilities Pre-Covid and Post-Covid Data source: Elaboration in Python



The most interesting observations can be done on the differences in the weights in the two periods. The weights “Pre-Covid” are reported in Figure 23, while the weights “Post-covid” are reported in Figure 22. On one hand, after the Covid crisis, the HPR remained well diversified shifting more weights on the Japan and Vietnam ETFs and decreasing a little bit the weights allocated to Canada and Australia. This happened because Japan experienced a relative little increase in volatility, as opposed to Australia which saw a huge increase. On the other hand, the GMV portfolio became even more concentrated. It allocates a big part of wealth to Japan, and the remaining is distributed among U.S., South Africa and Vietnam. Therefore, the GMV, with respect to the “Pre-Covid” period, discards from the allocation the ETFs tracking France, U.K., Canada and Australia. The GMV ended up investing in only 4 assets out of 10. The GMV constrained shifted a percentage of wealth from Australia to Japan and Vietnam. The common consequences in the allocations of each portfolio are the big drop in the weights allocated to Australia and the big increase in the weights allocated to Japan and Vietnam.

Figure 23 - Weights in Pre-Covid

Data source: Elaboration in Python

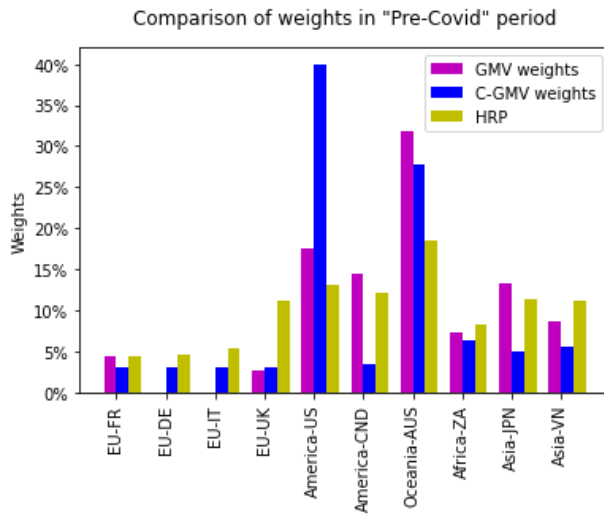
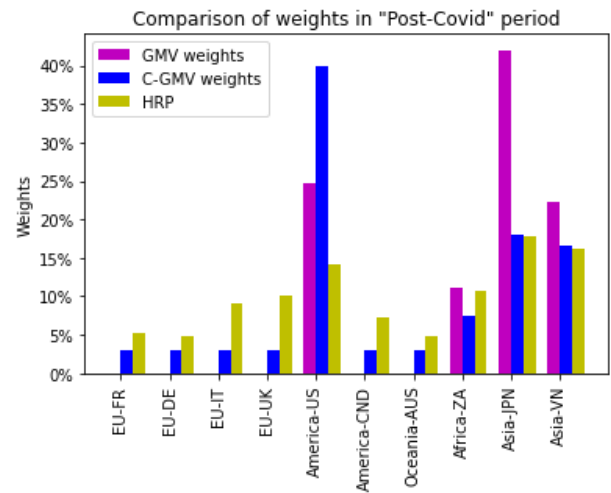


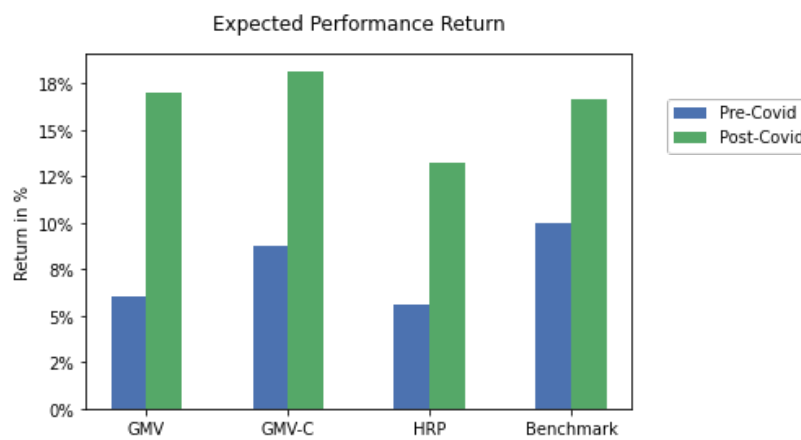
Figure 22 - Comparison of weights in Post-Covid

Data source: Elaboration in Python



In Figure 24, the expected annual performance returns delivered by each portfolio for each period are plotted. In the “Pre-Covid” subperiod, the highest expected return is delivered by the benchmark and it is of 10%. Again, the HRP delivers the poorest performance with an expected return of 5.6%. The GMV has an expected return of 6.1% and the GMV constrained of 8.7%. In the “Post-Covid” period the expected returns increase a lot. The GMV constrained is the best with an 18% expected return and the HRP delivers the lowest return of 13.3%.

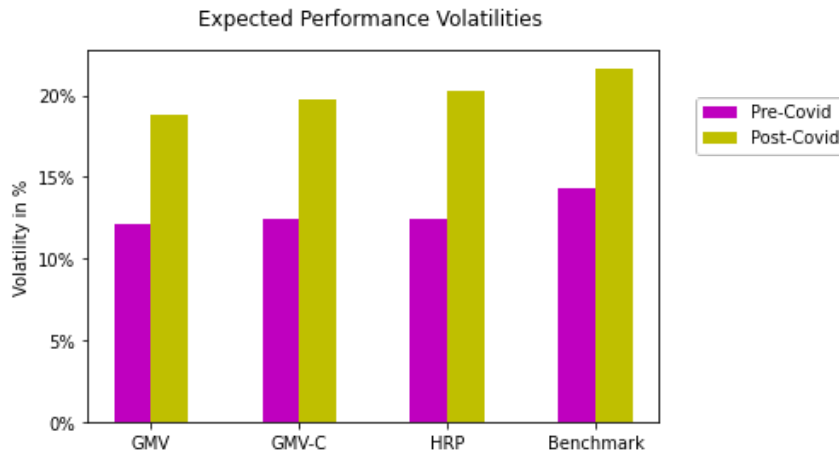
Figure 24 - Comparison of expected performance returns Pre-Covid and Post-Covid Data source: Elaboration in Python



The expected annual performance volatilities increased a lot in the second subperiod, as a consequence of the increase in the volatilities of the single assets. In the first period, the

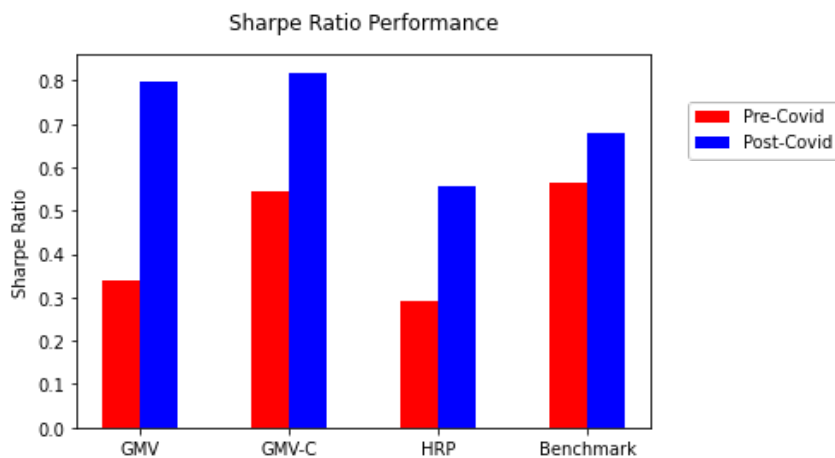
volatilities expected from the different portfolios are almost equal; in the “Post-Covid” period, the GMV achieves the lowest volatility. In the second subperiod, the volatilities are $\sigma_{HRP} = 20.24\%$, $\sigma_{GMV} = 18.8\%$, $\sigma_{GMV-C} = 19.74\%$ and $\sigma_{benchmark} = 21.65\%$.

Figure 25 - Comparison of expected performance volatilities Pre-Covid and Post-Covid Data source: Elaboration in Python



The resulting Sharpe ratios increased from the first to the second period. In particular, according to this measure of performance, in the “Pre-Covid” period the benchmark is the best investment because it achieves a Sharpe ratio of 0.56, the highest one. In the “Post-Covid” period, the GMV constrained is the portfolio that performs better, it allows to achieve a Sharpe ratio of 0.82. In both periods, the HRP delivers the lowest indexes, respectively, 0.29 and 0.56.

Figure 26 - Comparison of Sharpe Ratios Pre-Covid and Post-Covid Data source: Elaboration in Python



4.3.3 In-sample versus out-of-sample

The third analysis is done out-of-sample using the rolling approach with monthly rebalancing periods. The results are compared with the in-sample analysis, which corresponds to the first analysis implemented on the entire sample.

Looking at the track records of weights for the GMV portfolio and for the GMV constrained, respectively in Figure 28 and Figure 27, it is possible to observe how the allocations are quite concentrated over time. Furthermore, the allocations are not stable but constantly vary over time. This implies a certain amount of costs to face for the turnover. For example, the GMV allocation in the middle period contains the Germany ETF, while at the beginning and at the end it allocates zero weight to that ETF. In particular, with the spread of the Covid around February 2020, the allocations are completely reviewed with large movements, especially for the GMV portfolio. This means that, given the changes in the inputs, the resulting weights change consistently. The weight allocated to Australia (in blue) is suddenly close to zero, while the weight allocated to Japan (in red) sees a big increase going to 40%. Again, the allocation of the constrained GMV is guided by the lower bound of 40% on the U.S. and the upper bound on the emerging countries.

Figure 28 - Evolution of GMV weights out-of-sample Data source: Elaboration in Python

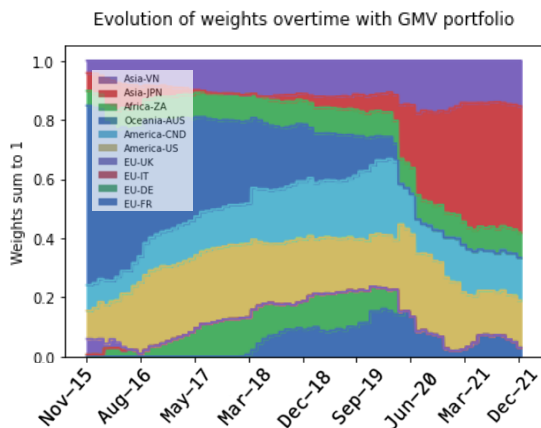
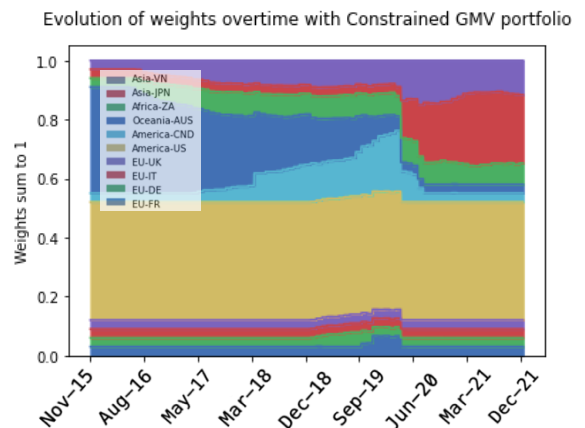


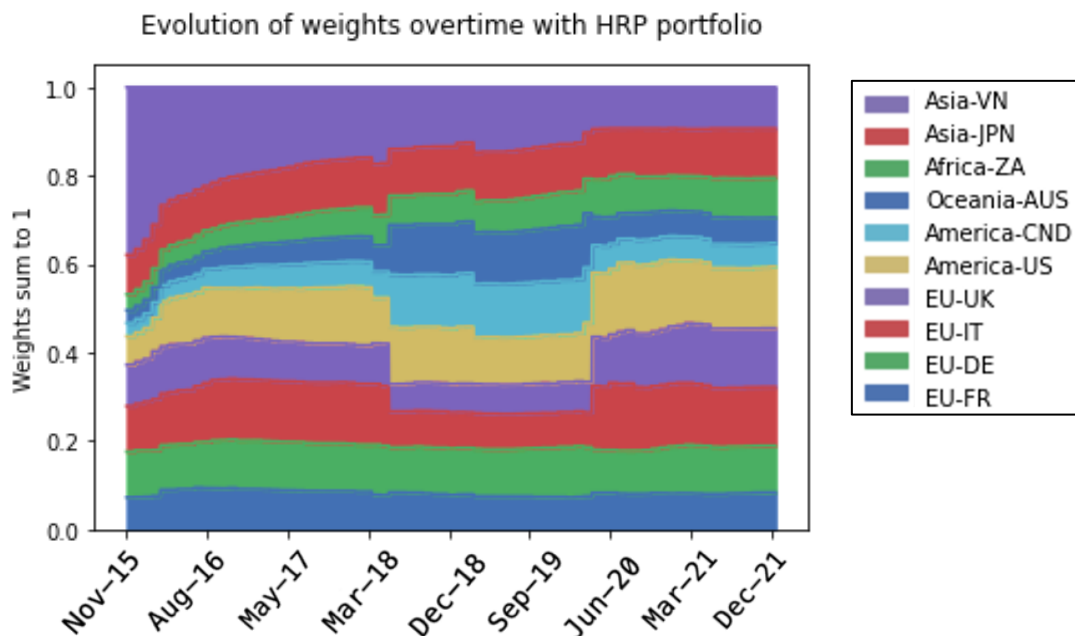
Figure 27 - Evolution of GMV constrained weights out-of-sample Data source: Elaboration in Python



The track records of the HRP in Figure 29 highlights the constant high diversification between assets and the higher stability of weights across time. The allocation is clearly more stable with respect to the other two portfolios. It doesn't experience relevant adjustments, meaning that the portfolio is well prepared to face changes in the volatilities' conditions. Given these

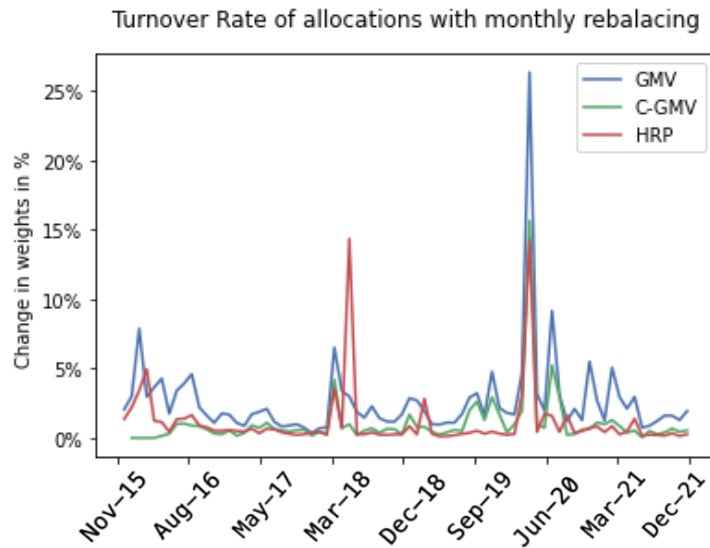
characteristics of the track records, the HRP seems to present a lower turnover and so lower rebalancing costs

Figure 29 - Evolution of HRP weights out-of-sample Data source: Elaboration in Python



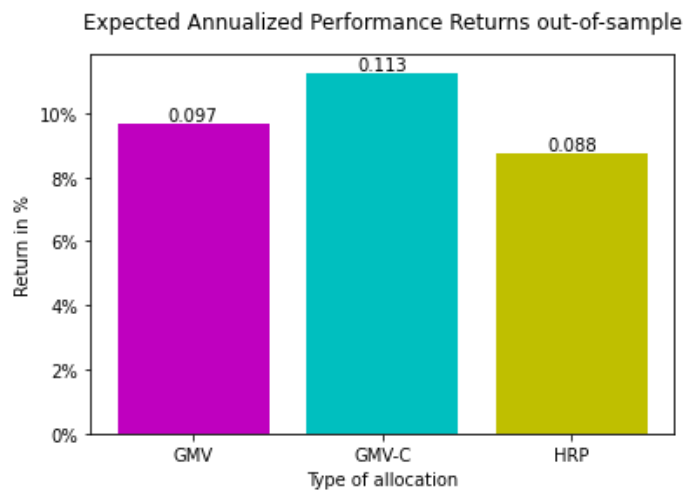
To verify the implication derived from the evolution of portfolio allocations, the turnover ratio is computed according to the formula $\frac{1}{2} |w_t - w_{t-1}|' \mathbf{1}_n$. The difference between the current and the previous weight is divided by 2 in order to consider only the effective turnover and avoid monitoring the total amount of trades (buy and sell). The evolution of the turnover indexes over time is represented in Figure 30. The turnover of the HRP goes up to 14.36% while, the maximum turnover for the GMV and the constrained GMV is respectively 26.35% and 15.63%. Furthermore, in most rebalances the turnover is lower for the HRP, especially compared to the GMV. This implies lower rebalancing costs achieved with the implementation of the HRP. There are two moments where the turnovers spike, one is around the month of March 2018 and the other around the end of February 2020. The first corresponds to a change in volatilities probably derived from Trump's announcement about the imposition of customs duties on products from China and from the increase in interest rates by the FED. This is reflected also in the weights' evolution. While the second one corresponds to the change in volatilities derived from the spread of Covid.

Figure 30 - Comparison of turnovers out-of-sample Data source: Elaboration in Python



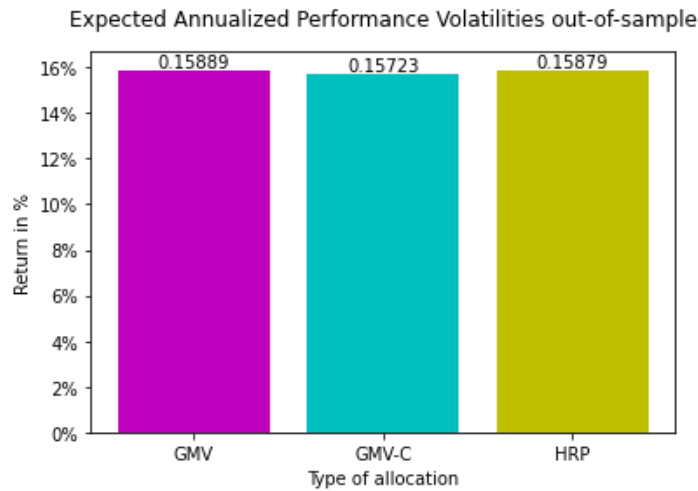
The expected performance returns out-of-sample are higher with respect to the ones in-sample. The constrained GMV portfolio delivers the higher expected return of 11.3%, while the HRP delivers only 8.8%. As can be deduced from Figure 31, the proportions between the different expected returns are equal to the in-sample case.

Figure 31 - Comparison of performance returns out-of-sample Data source: Elaboration in Python



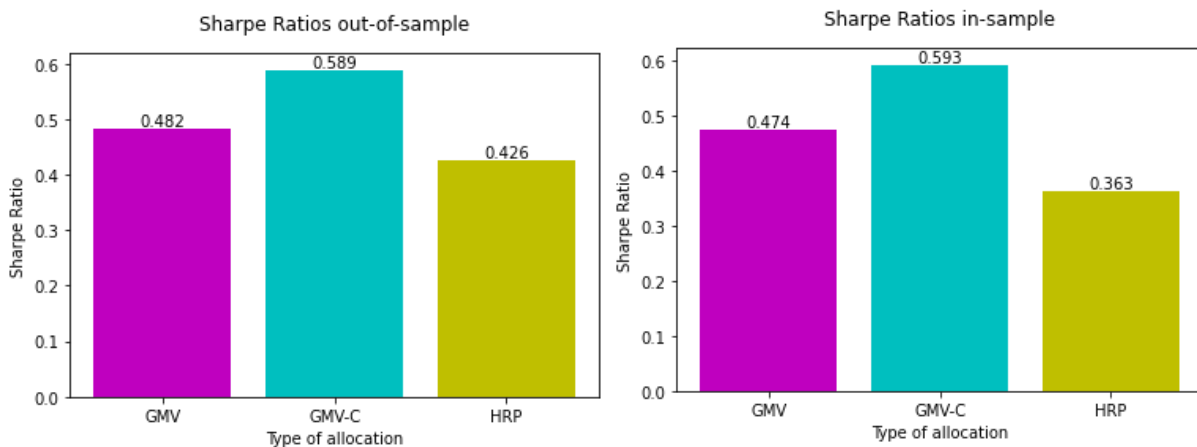
Even the expected volatilities out-of-sample are higher with respect to the ones in-sample. The volatility out-of-sample for the HRP is 15.879%, infinitesimally lower than the volatility of the GMV portfolio, which corresponds to 15.889%. The constrained GMV presents the lowest volatility with a value of 15.7%. The differences are really small, so it is possible to conclude that the models deliver similar volatilities.

Figure 32 - Comparison of performance volatilities out-of-sample Data source: Elaboration in Python



The Sharpe ratios out-of-sample for the GMV and the HRP are higher than the ratios in-sample, as can be seen in Figure 33. The Sharpe ratio for the HRP increases quite a lot out-of-sample with respect to the relative increase of the other portfolios. It increases by 17% with respect to the in-sample case, while the Sharpe ratio of the GMV increases only by 1.7% and the one of the GMV constrained decreases by 0.7%. However, the HRP remains the portfolio delivering the lowest Sharpe ratio, $SR_{HRP} = 0.43$. The highest value of the performance index is achieved by the constrained GMV portfolio, $SR_{GMV-C} = 0.59$. The baseline GMV delivers a Sharpe ratio of 0.48. Hence, the analysis out-of-sample identifies the GMV constrained as the best portfolio.

Figure 33 - Comparison of Sharpe ratios out-of-sample and in-sample Data source: Elaboration in Python



In Table 3 some descriptive statistics of the distribution of returns out-of-sample are reported. The mean corresponds to the previously plotted out-of-sample performance return. The HRP

has the lowest mean and the lowest median return, while the GMV constrained has the highest ones. The standard deviations correspond to the previously plotted performance volatilities out-of-sample, which are very similar to each other. The skewness is negative for all the portfolios; this means that the left tail is longer and the values are more concentrated on the right tail, as indicated by the relation between the mean and the median. The kurtosis is really high indicating leptokurtic distributions. This condition denotes that extreme values, with respect to the normal distribution, are more likely to occur, in particular for the HRP.

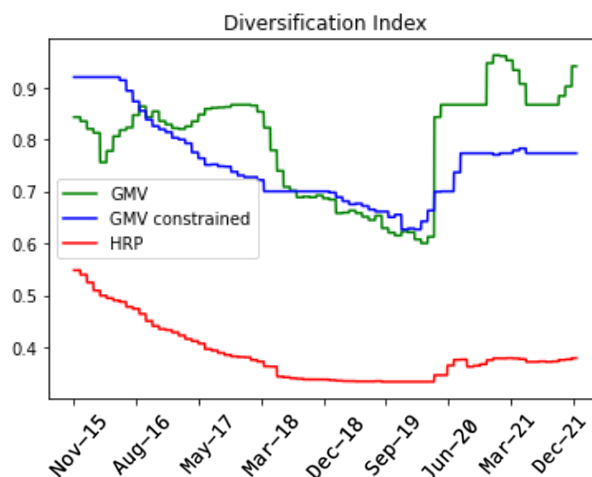
Table 3 - Descriptive statistic of the returns out-of-sample Data source: Elaboration in Excel

Descriptive statistic of the distribution of expected performance returns out-of-sample

	Mean	Median	Variance	Std Dev	Skewness	Kurtosis
GMV	0.10	0.15	0.03	0.16	-1.38	15.31
C-GMV	0.11	0.23	0.03	0.16	-1.24	13.13
HRP	0.09	0.13	0.03	0.16	-1.51	18.01

In order to confirm the previous intuitions, the diversification index is calculated. It is computed as the deviation of the allocations with respect to the weights computed for the equally weighted portfolio. So, in the perfect diversification case, the diversification index takes value equal to zero. The index reflects the diversification already seen in the track records of weights and it is reported in Figure 34. The HRP is the most diversified and it reaches values close to zero. The constrained GMV portfolio is more diversified across time than the GMV, due to the imposed bounds. The GMV reaches very concentrated distributions, especially when the diversification index goes up close to 1.

Figure 34 - Diversification Indexes out-of-sample Data source: Elaboration in Python



4.4 Conclusions

The results delivered from the empirical application lead to some concluding remarks.

The first interesting observation is how the different portfolios react when facing shocks. In general, the allocation of weights delivered by the GMV portfolio is concentrated, while the allocation of the HRP is always diversified. A drawback of the concentrated distribution of the GMV is the exposition to idiosyncratic shocks. Indeed, when the volatility of the Australia ETF experienced a high increase, the weight allocated to it dramatically goes close to zero. Also the exposition to systemic shocks is high. For example, with the spread of the Covid and the Brexit, all countries experienced a shock in the prices, even if some more than others. Looking at the change in weights from the “Pre-Covid” to the “Post-Covid” period, it is evident how the allocation for the GMV portfolio became even more concentrated by dropping the investments in France, U.K., Canada and Australia and allocating the wealth to only 4 assets out of 10. The GMV presents in fact a diversification index close to 1, meaning that its distribution is highly concentrated. The increased concentration exposes even more the portfolio to idiosyncratic shocks, leading problematic situations to have a big negative impact. The same observations can be done by looking at the track records of the weights over time. The GMV changes in a more irregular way its allocation and, in particular with the spread of the Covid, the weights allocated are completely reviewed. This implies that given changes in the inputs the resulting weights change consistently, implying higher turnover and higher rebalancing costs. This is a consequence of the high condition number of the covariance matrix of the ETFs’ returns which corresponds to 220; indeed, the results delivered by the quadratic optimizers are expected to be unstable. As the GMV portfolio, also the constrained GMV presents an allocation that varies quite frequently over time, except for the weights allocated to the ETFs where upper and lower bounds are imposed.

The HRP always delivers highly diversified allocations. Looking at the out-of-sample case, the allocation is really stable over time. This comes from the constant and meaningful diversification implemented and from the fact that the HRP considers the correlations between the different assets, so, in case of shocks, it does not vary its allocation in an irregular way because it is already prepared. The more diversified distribution of the HRP ensures less negative effect from idiosyncratic shocks, as well as systemic. This first set of observations confirms the findings of López De Prado (2016): the HRP guarantees better protection against

idiosyncratic and systemic risks. Furthermore, the more stable allocation of the HRP allows to achieve a lower turnover and so lower rebalancing costs.

The next aspect to consider is the volatility delivered by the different models in different settings. In the in-sample case, the GMV portfolio meets its objective of achieving the minimum volatility. However, it presents a volatility of 14.1%, only 0.2% lower with respect to the volatility of 14.3% achieved by the HRP. Similarly, in the “Pre-Covid” subperiod, the standard deviation of the HRP is higher, but very close to the one of the GMV portfolio. The same happens in the “Post-Covid” period, the HRP delivers a volatility of 20.23% close to the 18.8% volatility of the GMV. However, in the out-of-sample analysis, the HRP delivers a volatility that is infinitesimally inferior to the GMV. This result does not show an outstanding improvement done by the HRP; however, it is in line with the results of the Monte-Carlo simulation developed by López De Prado (2016). Among the different techniques implemented, the constrained GMV reaches the lowest volatility in the majority of the cases, this is probably due to the fact that the imposition of bounds allows the portfolio to reach a higher diversification, together with its own objective of minimizing the volatility.

Using the Sharpe ratio as performance measure for evaluating the different strategies allows to identify the best portfolio. In the entire sample case and in the “Pre-Covid” scenario, the benchmark and the constrained GMV portfolio deliver the highest performances. In the “Post-Covid” period the GMV portfolio and, again, the constrained GMV reach the highest Sharpe ratios. In the out-of-sample analysis, the constrained GVM is the best, as in most cases. This means that the bounds have been set with a meaningful logic and allow to achieve greater results, probably thanks to the better diversification, as previously highlighted. The HRP does not outperform the other strategies under any circumstances, according to the Sharpe Ratio. However, from the in-sample to the out-of-sample case, it experiences a major improvement in the Sharpe ratio reaching a value of 0.43, really close to the one delivered by the GMV, $SR_{GMV} = 0.48$. This is an indication of how much the HRP improves its performance out-of-sample.

The HRP in this case does not outperform the quadratic optimizers considered, even if it reaches greater weights’ diversification and stability. The possible reasons are many. First of all, both the Markowitz’s portfolios are improved through the imposition of the positivity constraint and the lower and upper bounds. The constraints allow to achieve more diversification and better performances, and this is a potential reason for the better results delivered by the quadratic

optimizers. The dataset considered does not include a large number of assets and as highlighted by López De Prado (2016), the larger the investment opportunities, the greater are the improvements that the HRP allows to achieve with respect to the quadratic optimizers. This comes from the effect of the size of the investment universe on the condition number of the covariance matrix. A higher number of assets would imply a more ill-conditioned matrix and so more unstable results. Furthermore, the correlation structure of the dataset is not so strong because the ETFs are already really diversified and the only big shock that occurs from 2011 to 2021 is the spread of the Covid in 2020. These could be other potential reasons for the lack of outstanding results of the HRP, because, as studied by López De Prado (2016, p.11), “HRP’s out-of-sample outperformance becomes even more substantial ... when more shocks are added or a stronger correlation structure is considered”. This is due to the fact that more shocks would highlight the better capacity of the HRP to react to idiosyncratic and systemic issues, while a stronger correlation would imply a higher condition number, so higher benefits from the lack of matrix’s inversion.

CONCLUSION

In financial applications, portfolio allocation has always been really important. Markowitz developed the traditional model that provides the optimal allocation based on the mean-variance criterion and the diversification principle. During the years, different pitfalls of quadratic optimizers have emerged. However, portfolio allocation models are in continuous evolution, especially considering the new application of machine learning techniques, indeed a model that overcomes the instability of results, typical of quadratic optimizers, has been elaborated. The model is the Hierarchical Risk Parity, recently developed by López De Prado. The risk parity approach and the clustering of assets allow to achieve a more meaningful diversification over time, which is more stable in time of crisis. Furthermore, the lack of inversion of the covariance matrix brings greater stability in the computed weights.

I applied the model to real data, instead of simulated data, via Python. I implemented it on a set of equity ETFs, geographically diversified. The first evidence is that the clustered correlation matrix allows achieving a lower condition number, which means higher stability in the results. This implementation would be beneficial also for Markowitz's model, given that the inversion of ill-conditioned matrices leads to huge instability. The second evidence is the better performance of the HRP with respect to the traditional Global Minimum Variance, in terms of stability, diversification and turnover. In case of shocks, both idiosyncratic and systemic, the HRP slightly adjusts the weights; the GMV instead is highly concentrated and in case of shocks, completely reviews the allocation, taking extreme positions. So, the greater stability of the HRP in the weights' allocation with respect to the quadratic optimizers, especially in crisis time, is shown. This implies that changes in the inputs do not lead to big changes in the output, therefore the intrinsic instability of quadratic optimizers is eliminated. In this way the reliability of the investment value of the portfolio computed increases. This feature could be furtherly improved by implementing the dissimilarity measure based on the negative log of lower tail dependence coefficient, proposed by Lohre, Rother and Schäfer (2020). Another consequential advantage of the HRP is the low turnover which implies facing a lower amount of rebalancing costs. The third evidence is that in the out-of-sample analysis the HRP achieves lower risk, even if the objective of the GMV is to minimize the volatility; this could be particularly beneficial for risk parity investors that frequently implement leveraging. The last evidence is that the Sharpe ratio delivered by the HRP out-of-sample is slightly lower than the one of the GMV, even if the volatility is lower. This could be a consequence of the small size of the dataset, of the weak correlation scheme or of the small number of shocks that occurred in the selected period of time.

APPENDIX

A. Efficient Frontier from Markowitz's framework

A.1 Efficient Frontier without risk-free asset

Given the setting, a portfolio contained in the entire set of portfolios Δ_N is defined by:

- the random return $\tilde{r}_w = \sum_{n=1}^N w_n \tilde{r}_n$;
- the expected value of the random return $\mathbb{E}[\tilde{r}_w] = w^T e = \sum_{n=1}^N w_n e_n$;
- the variance of the random return $\sigma^2(\tilde{r}_w) = w^T V w = \sum_{n=1}^N \sum_{m=1}^N w_n w_m \text{Cov}(\tilde{r}_n, \tilde{r}_m)$.

The condition $w \in \Delta_N$ can be expressed as $w^T \mathbf{1} = \mathbf{1}$, where $\mathbf{1} := (1, \dots, 1)^T \in \mathbb{R}^N$.

An efficient portfolio could be the one that for a given level of risk maximizes the expected return μ . In this case there would be two constraints: one relative to the target risk and the admissibility constraint. The mathematical representation would be

$$\max_{w \in \mathbb{R}^N} w^T e,$$

under the constraints

$$w^T V w = \bar{\sigma}^2 \quad \text{and} \quad w^T \mathbf{1} = 1.$$

When elaborating the solution w^* to the optimization problem, the notation of the scalar quantities (expresses in matrices) is introduced as

$$\begin{aligned} g &:= \frac{BV^{-1}\mathbf{1} - AV^{-1}e}{D}, & h &:= \frac{CV^{-1}e - V^{-1}\mathbf{1}}{D}, \\ A &:= \mathbf{1}^T V^{-1} e, & B &:= e^T V^{-1} e, & C &:= \mathbf{1}^T V^{-1} \mathbf{1}, & D &:= BC - A^2. \end{aligned}$$

Alternatively, the optimal solution can be equivalently found by minimizing $w^T V w / 2$ through the first-order optimality conditions derived for the minimization of the Lagrangian function:

$$L(w, \lambda, \gamma) := \frac{1}{2} w^T V w + \lambda(\mu - w^T e) + \gamma(1 - w^T \mathbf{1}).$$

The explicit solution will be

$$w^* = \lambda V^{-1} e + \gamma V^{-1} \mathbf{1},$$

consequently, replacing the scalar quantities, it is obtained

$$\lambda = \frac{C\mu - A}{D},$$

$$\gamma = \frac{B - A\mu}{D},$$

$$w^* = g + h\mu.$$

Considering the objective, the E-V rule and the preferences of the investor, the relevant quantities of the portfolio P are only

$$\mu_p = \mathbb{E}[\tilde{r}_w] = w^T e = \sum_{n=1}^N w_n e_n;$$

$$\sigma_p^2 = \sigma^2(\tilde{r}_w) = w^T V w = \sum_{n=1}^N \sum_{m=1}^N w_n w_m \text{Cov}(\tilde{r}_n, \tilde{r}_m).$$

The different efficient portfolios identified are not always comparable, therefore, to order the different investment opportunities the preferences of the agents must be considered through the utility function. Even the utility function must be expressed in terms of mean and variance:

$$\mathbb{E}[U(W_1)] = \mathbb{E}[W_1] - \lambda \mathbb{V}[W_1], \quad W_1 = W_0(1 + r_p)$$

$$\mathbb{E}[U(W_1)] \simeq W_0(\mathbb{E}[r_p] - \lambda W_0 \mathbb{V}[r_p])$$

$$\mathbb{E}[U(W_1)] \propto \mathbb{E}[r_p] - \eta \mathbb{V}[r_p]^{41}$$

The η is the risk aversion coefficient on which depends the slope of the indifference curves. The aim of the agents is to maximize their expected utility function, considering their initial wealth:

$$\max_w \mathbb{E}[U(W_1)],$$

which is equivalent to

$$\max_w \mathbb{E}[r_p] - \eta \mathbb{V}[r_p].$$

The utility function is expressed in terms of mean and variance and it can be defined as:

$$U = \mu_p - \frac{\eta}{2} \sigma_p^2,$$

where η is the risk aversion coefficient. To find the optimal choice, the utility function must be maximized given the admissibility constraint:

$$\max_{w \in \mathbb{R}^N} U = w^T e - \frac{\eta}{2} w^T V w$$

under the constraint

$$w^T \mathbf{1} = 1.$$

The solution is $w^* = \frac{A}{\eta} w_T - \frac{A-\eta}{\eta} w_{GMV}$. Considering that the $w \lim_{\eta \rightarrow \infty}$ converges to w_{GMV} , it means that an infinitely risk adverse agent will have as main objective the reduction of risk, so of volatility, ending up allocating his wealth according to the GMV weights. In fact, the

⁴¹ This holds for a quadratic utility or for a negative exponential utility and Gaussian returns.

indifference curves will be steeper and so the tangency with the EF will be located graphically nearer to the GMV portfolio. However, taking into consideration $\eta \rightarrow 0$, the agent will leverage his portfolio by going short on w_{GMV} and long on w_T , because, in this case, the agent is willing to take more risk.

A.2 Efficient Frontier with risk-free asset

In the setting,

- $\mathbb{E}[\tilde{r}_n] \neq r_f > 0$ for at least one $n \in \{1, \dots, N\}$;
- $e = (e_1, \dots, e_N)^T \in \mathbb{R}^N$ the vector of expected returns and
- $V \in \mathbb{R}^{N \times N}$ the variance-covariance matrix of the random vector \tilde{r} , which is assumed to be positive definite.

Given the solution $w^* = \frac{\mu - r_f}{K} V^{-1}(e - r_f \mathbf{1})$, K is defined as

$$K = (e - \mathbf{1}r_f)^T V^{-1}(e - \mathbf{1}r_f) = B - 2Ar_f + Cr_f^2 > 0.^{42}$$

As in the previous case, the optimal solution can be equivalently found through the first-order optimality conditions derived for the minimization of the Lagrangian function:

$$L(w, \lambda) = \frac{1}{2} w^T V w + \lambda(\mu - w^T e - (1 - w^T \mathbf{1})r_f).$$

The solution will be

$$w^* = \lambda V^{-1}(e - r_f \mathbf{1}),$$

consequently, it is obtained

$$\lambda = \frac{\mu - r_f}{K}, \text{ with } K > 0.$$

The portfolio w^* will have variance defined as

$$\sigma^2(\tilde{r}_{w^*}) = \frac{(\mathbb{E}[\tilde{r}_{w^*}] - r_f)^2}{K}.$$

Considering that the objective is to identify the optimal investment decision for a rational agent, the utility function must be recalled:

$$U = \mu_p - \frac{\eta}{2} \sigma_p^2.$$

⁴² The scalars A and B are computed as in the previous case.

B. Further Extensions of the HRP model

Regarding the covariance matrix, for example, it can be estimated with the sample-based method, with the exponentially weighted moving average (EWMA) or with the dynamic conditional correlation GRACH (DCC-GARCH). The authors Jain, P. and Jain S. (2019) tried to identify the estimator of covariance which allows reaching better performance for each type of portfolio, included the traditional portfolios. The authors performed the Superior Predictive Ability (SPA) test of Hansen to identify if the better performances are significant or not. It emerged that with the application of the DDC-GARCH (1,1), the portfolios always significantly reach better performance. In particular, the HRP seems to be the best for minimizing the out-of-sample variance and the expected shortfall when the DCC GARCH is used for the covariance.

Another point of view on the estimation of the covariance matrix is introduced by Lohre, Rother and Schäfer (2020). The authors recognized that correlations could increase significantly during financial crises because of contagion effects. Therefore, it could be useful to focus on the correlation among assets during times of large losses by using the lower tail dependence coefficient (LTDC) which implies a better ability to diversify during a crisis. In the study, the HRP is analyzed by carrying out the clustering based on tail dependences correlation instead of the standard correlation. To estimate tail dependence coefficients a CSR-estimator (Conditional Spearman's Rho) is used, and the dissimilarity measure used is the negative log of LTDC. The modified HRP strategy seems to have interesting diversification properties and it allows a better tail risk management.

Regarding the approach for splitting the weights, the authors Katzke and Pfitzinger (2019) developed an improved HRP. In this model the hierarchical clustering is done through the divisive analysis (DIANA) with a top-down approach, obtaining a model labeled DHRP. An additional modification is done to elaborate an ulterior model. In the quasi-diagonalization step an optimization problem it is introduced and to solve it the genetic permutation algorithm is used, obtaining the model labeled GHRP. In the recursive bisection stage τ is introduced as a measure of how much the bisection follows the shape of the dendrogram. In López De Prado (2016), where τ is equal to 0, the "naive allocation" is performed, while if τ is equal to 1 the "dissimilarity-based allocation" is carried out. The result from a backtest is that the top-down approach and the use of the genetic permutation algorithm together with the dissimilarity-based

allocation allow to reach lower out-of-sample volatility, better risk-adjusted returns and lower turnover.

Raffinot (2018) merges together different modifications of the HRP. By merging the Hierarchical Clustering Asset Allocation model and the Hierarchical Risk Parity, the author created a new model called Hierarchical Equal Risk Contribution (HERC). It implements the identification of an optimal number of clusters based on the Gap index, the top-down recursive bisection based on the shape of the dendrogram, the equal risk contribution allocation and within each cluster the naive risk parity allocation. Using different performance measures the results show how the HERC portfolio based on downside risk measures, especially based on the CDaR (Conditional Drawdown at Risk), reaches better risk-adjusted performances and better diversification of risk capital with respect to other traditional portfolios.

Furthermore, HRP is a robust procedure which avoids the inversion of the covariance matrix and it could be used to substitute unstable econometric regression methods such as VAR (Vector AutoRegression) and VECM (Vector Error Correction Model). In future, it could be also applied to optimization processes, factor models, regression analysis and econometrics in general.

C. Python codes

C.1 Source Code for the HRP model

Source code of López De Prado (2016) for the HRP applied to the selected dataset.

```
# Hierarchical Risk Parity
import matplotlib.pyplot as plt
import scipy.cluster.hierarchy as sch, random
from scipy.spatial.distance import squareform, pdist
from scipy.cluster.hierarchy import linkage, dendrogram
import numpy as np
import pandas as pd
from pypfopt.efficient_frontier import EfficientFrontier
from pypfopt import risk_models
from pypfopt import expected_returns
from pypfopt import plotting
from pypfopt import objective_functions
from pypfopt import HRPopt
# Read prices in data
prices = pd.read_csv("/Users/ilariapadovan/Desktop/Tesi_PythonCode/Dataset.csv",
parse_dates=True, index_col="date")
#change name of columns
prices.columns=['World', 'EU-FR', 'EU-DE', 'EU-IT', 'EU-UK', 'America-US', 'America-
CND', 'Oceania-AUS', 'Africa-ZA', 'Asia-JPN', 'Asia-VN']
# Compute daily returns
ret = prices.pct_change(1)
returns= ret.iloc[:,1:]
lab1=['EU-FR', 'EU-DE', 'EU-IT', 'EU-UK', 'America-US', 'America-CND', 'Oceania-
AUS', 'Africa-ZA', 'Asia-JPN', 'Asia-VN']
# Compute covariance of returns
cov=returns.cov()
# TREE CLUSTERING
# Computation of the correlation matrix
corr=returns.corr()
# Plot correlation matrix
plt.figure()
plotting.plot_covariance(corr)
plt.title('Correlations of daily returns in period 2011-2021',pad=20)
# Computation of the distance matrix
def correlDist(corr):
    # A distance matrix based on correlation, where 0<=d[i,j]<=1
    # This is a proper distance metric
    dist=((1-corr)/2.)**.5 # distance matrix
    return dist
d_corr=correlDist(corr)
# Computation of the linkage matrix using the Euclidean distance measure and the
single linkage method
link_corr=sch.linkage(d_corr, 'single', 'euclidean')
# Plot the dendrogram
```

```

d=dendrogram(link_corr,labels=lab1,leaf_rotation=45)
plt.title('Clusters derived from the tree clustering', pad=15)
#QUASI-DIAGONALIZATION
def getQuasiDiag(link):
    # Sort clustered items by distance
    link=link.astype(int)
    sortIx=pd.Series([link[-1,0],link[-1,1]])
    numItems=link[-1,3] # number of original items
    while sortIx.max()>=numItems:
        sortIx.index=range(0,sortIx.shape[0]*2,2) # make space
        df0=sortIx[sortIx>=numItems] # find clusters
        i=df0.index;j=df0.values-numItems
        sortIx[i]=link[j,0] # item 1
        df0=pd.Series(link[j,1],index=i+1)
        sortIx=sortIx.append(df0) # item 2
        sortIx=sortIx.sort_index() # re-sort
        sortIx.index=range(sortIx.shape[0]) # re-index
    return sortIx.tolist()
#Ordered matrix
sortIx=getQuasiDiag(link_corr)
sortIx=corr.index[sortIx].tolist() # recover labels
sort_corr=corr.loc[sortIx,sortIx] # reorder
#Plot the ordered matrix
plt.figure()
plotting.plot_covariance(sort_corr)
plt.title('Ordered Correlations of daily returns in period 2011-2021',pad=20)
#Compute the condition number of the clustered covariance matrix
print (np.linalg.cond(sort_corr, p=None))
#RECURSIVE BISECTION
# Define cluster variance
def getIVP(cov,**kargs):
    # Compute the inverse-variance portfolio
    ivp=1./np.diag(cov)
    ivp/=ivp.sum()
    return ivp
def getClusterVar(cov,cItems):
    # Compute variance per cluster
    cov_=cov.loc[cItems,cItems] # matrix slice
    w_=getIVP(cov_).reshape(-1,1)
    cVar=np.dot(np.dot(w_.T,cov_),w_)[0,0]
    return cVar
# Define recursive bisection
def getRecBipart(cov,sortIx):
    # Compute HRP allocation
    w=pd.Series(1,index=sortIx)
    cItems=[sortIx] # initialize all items in one cluster
    while len(cItems)>0:
        cItems=[i[int(j):int(k)] for i in cItems for j,k in ((0,len(i)/2),
(len(i)/2,len(i))) if len(i)>1] # bi-section
        for y in range(0,len(cItems),2): # parse in pairs
            cItems0=cItems[y] # cluster 1
            cItems1=cItems[y+1] # cluster 2
            cVar0=getClusterVar(cov,cItems0)

```

```

        cVar1=getClusterVar(cov,cItems1)
        alpha=1-cVar0/(cVar0+cVar1)
        w[cItems0]*=alpha # weight 1
        w[cItems1]*=1-alpha # weight 2
    return w
#Compute weights
hrp=getRecBipart(cov,sortIx)
print(hrp)
#Plot weights
pd.Series(hrp).plot.pie(figsize=(10, 10))
#Performance expected volatility
def get_vol(self, data, weights):
    # normal volatility calculation
    cov = data.cov().values
    weights = weights.values
    var = np.dot(weights.T.dot(cov), weights)
    return np.sqrt(var * 252)
v=get_vol('self',returns, hrp)
print (v)
#Performance expected return
mu = ret.mean()
mu=((mu.add(1).pow(252)).sub(1))
r=np.dot((np.array(mu[1:])),np.array(hrp))
print (r)

```

C.2 Code snippet for the empirical application

```

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from pypfopt.efficient_frontier import EfficientFrontier
from pypfopt import risk_models
from pypfopt import expected_returns
from pypfopt import plotting
from pypfopt import objective_functions
from pypfopt import HRPOpt
from scipy import stats
import matplotlib.ticker as mtick
#Entire Sample
# Read in price data
data = pd.read_csv("/Users/ilariapadovan/Desktop/Tesi_PythonCode/Dataset.csv",
parse_dates=True, index_col="date")
#Restrict the sample to a common one
p = data.dropna(0) #price indices
#change name of columns
p.columns=['World','EU-FR','EU-DE','EU-IT','EU-UK','America-US','America-
CND','Oceania-AUS','Africa-ZA','Asia-JPN','Asia-VN']
# computing daily returns
r = p.pct_change(1)
#compute expected returns
mu = r.mean()

```



```

mu=((mu.add(1)).pow(252)).sub(1)
#plot expected returns
lab=['World', 'EU-FR', 'EU-DE', 'EU-IT', 'EU-UK', 'America-US', 'America-CND', 'Oceania-
AUS', 'Africa-ZA', 'Asia-JPN', 'Asia-VN']
fig = plt.figure()
ax = fig.add_subplot(1,1,1)
ax.yaxis.set_major_formatter(mtick.PercentFormatter(1.0))
plt.xticks(rotation='vertical')
ax.set_ylabel('Return in %')
ax.set_xlabel('ETFs labeled as geographic areas')
plt.title('Annualized historical average of daily returns in period 2011-2021',
pad=15)
plt.bar(lab,mu)
plt.show()
#compute and plot the covariance matrix
S = r.cov()
S=((S.add(1)).pow(252)).sub(1)
fig = plt.figure()
ax = fig.add_subplot(1,1,1)
ax.yaxis.set_major_formatter(mtick.PercentFormatter(1.0,decimals=0))
plt.xticks(rotation='vertical')
ax.set_ylabel('Volatility in %')
ax.set_xlabel('ETFs labeled as geographic areas')
plt.title('Annualized historical daily volatilities in period 2011-2021', pad=15)
plt.bar(lab,(np.sqrt(np.diag(S))), color='y')
#compute and plot the correlation matrix
S1=r.corr()
plt.figure()
plotting.plot_covariance(S1)
plt.title('Correlations of daily returns in period 2011-2021',pad=20)
#Condition number of the covariance matrix
print (np.linalg.cond(S, p=None))
#Efficient Frontier
ef = EfficientFrontier(mu[1:], S.iloc[1:,1:])
fig, ax = plt.subplots()
plotting.plot_efficient_frontier(ef)
ax.scatter((np.sqrt(np.diag(S)))[0], mu[0], marker="*", s=100, c="r", label="MSCI
WORLD ETF")
ax.yaxis.set_major_formatter(mtick.PercentFormatter(1.0,decimals=0))
ax.xaxis.set_major_formatter(mtick.PercentFormatter(1.0,decimals=0))
ax.legend()
plt.title('Efficient Frontier')
plt.show()
# Optimize for Minimum Volatility
ef = EfficientFrontier(mu[1:], S.iloc[1:,1:])
raw_weights_min=ef.min_volatility()
cleaned_weights_min= ef.clean_weights()
print (cleaned_weights_min)
fig, ax = plt.subplots()
plotting.plot_weights(cleaned_weights_min)
#check
#return manually calculated 'a'
a=np.dot((np.array(mu[1:])),(np.array(list(cleaned_weights_min.values()))))

```

```

print(a)
#volatility manually calculated 'b'
b=np.sqrt(np.dot(np.dot(np.array(list(cleaned_weights_min.values())),(S.iloc[1:,1:]
)), np.array(list(cleaned_weights_min.values()))))
print(b)
#Optimize with constrained GMV
#Group Constraints
#map each ticker to a geographic area
sector_mapper={'EU-FR':'Developed','EU-DE':'Developed','EU-IT':'Developed','EU-
UK':'Developed','America-US':'America','America-CND':'Developed','Oceania-
AUS':'Developed','Africa-ZA':'Emerging','Asia-JPN':'Developed','Asia-
VN':'Emerging'}
sector_lower = {"America":0.40}
sector_upper = {"Emerging": 0.25}
efC = EfficientFrontier(mu[1:], S.iloc[1:,1:],weight_bounds=(0.03, 1))
for sector in sector_upper:
    is_sector = [v == sector for k, v in sector_mapper.items()]
    # Sum of weights from that sector has to be less than the upper bound
    efC.add_constraint(lambda w:sum(w[is_sector]) <= sector_upper[sector])
for sector in sector_lower:
    is_sector = [v == sector for k, v in sector_mapper.items()]
    efC.add_constraint(lambda w:sum(w[is_sector]) >= sector_lower[sector])

pwgtGMV=efC.min_volatility()
fig, ax = plt.subplots()
plotting.plot_weights(pwgtGMV)
#manual check
RetCGMV=np.dot(np.array(mu[1:]),(np.array(list(pwgtGMV.values()))))
print(RetCGMV)
#Optimize with HRP
#Compute the weights
rets= r
hrp = HRPOpt(rets.iloc[:,1:])
hrp.optimize()
hrp_weights = hrp.clean_weights()
print(hrp_weights)
#Plot weights
pd.Series(hrp_weights).plot.pie(figsize=(10, 10),autopct='%1.1f%%')
#manual check
np.dot(np.array(mu[1:]),(np.array(list(hrp_weights.values()))))
#Benchmark
RetB=(mu[0])
print('Expected annual return:',RetB)
SB=(np.sqrt(np.diag(S)[0]))
print('Annual Volatility:',SB)
SrB=RetB/SB
print('Sharpe Ratio:',SrB)
#Comparison of weights
lab1=['EU-FR','EU-DE','EU-IT','EU-UK','America-US','America-CND','Oceania-
AUS','Africa-ZA','Asia-JPN','Asia-VN']
x = np.arange(len(lab1)) # the label locations
width = 0.25 # the width of the bars
fig = plt.figure()

```

```

ax = fig.add_subplot(1,1,1)
ax.yaxis.set_major_formatter(mtick.PercentFormatter(1.0,decimals=0))
ax.bar(x - width/2,cleaned_weights_min.values(), width, label='GMV: Global Minimum
Variance')
ax.bar(x + width/2,pwgtGMV.values(), width, label='C-GMV: Constrained Global
Minimum Variance')
ax.bar((x+width*1.5),hrp_weights.values(),width,label='HRP: Hierarchical Risk
Parity')
ax.set_ylabel('Weights')
ax.set_title('Portion of wealth in % allocated to each ETF', pad=15)
plt.xticks(ticks=x, labels=lab1)
plt.xticks(rotation='vertical')
ax.legend(bbox_to_anchor=(1.05,0.6),loc="lower left",prop={'size': 10})
plt.show()
#Compare return,volatility and Sharpe Ratio of of HRP,GMV, C- GMV and benchmark
rf=0.02
#GMV performance
RetGMV,SGMV,SrGMV=ef.portfolio_performance(verbose=True)
#GMV constrained performace
RetCGMV,SCGMV,SrCGMV=efC.portfolio_performance(verbose=True)
#HRP performance
RetHRP,SHRP,SrHRP=hrp.portfolio_performance(verbose=True)
#Benchmark performance
RetB=(mu[0])
print('Expected annual return:',RetB)
SB=(np.sqrt(np.diag(S)[0]))
print('Annual Volatility:',SB)
SrB=(RetB-rf)/SB
print('Sharpe Ratio:',SrB)
#Plot returns
lab1=['GMV', 'GMV-C', 'HRP', 'Benchmark']
fig=plt.figure(figsize=(7, 5))
ax = plt.subplot(1,1,1)
p1=ax.bar(lab1, (RetGMV,RetCGMV,RetHRP,RetB),color=['m','c','y','g'])
ax.yaxis.set_major_formatter(mtick.PercentFormatter(1.0,decimals=0))
ax.set_ylabel('Return in %')
ax.set_xlabel('Type of allocation')
ax.bar_label(p1,fmt='%.3f')
ax.set_title('Expected Annualized Performance Returns in-sample', pad=15)
#Plot volatilities
lab1=['GMV', 'GMV-C', 'HRP', 'Benchmark']
fig=plt.figure(figsize=(7, 5))
ax = plt.subplot(1,1,1)
p2=ax.bar(lab1, (SGMV,SCGMV,SHRP,SB),color=['m','c','y','g'])
ax.yaxis.set_major_formatter(mtick.PercentFormatter(1.0,decimals=0))
ax.set_ylabel('Volatility in %')
ax.set_xlabel('Type of allocation')
ax.set_title('Expected Annualized Performance Volatilities in-sample', pad=15)
ax.bar_label(p2,fmt='%.3f')
#Plot Sharpe ratios
lab1=['GMV', 'GMV-C', 'HRP', 'Benchmark']
fig=plt.figure(figsize=(7, 5))
ax = plt.subplot(1,1,1)

```

```

p3=ax.bar(lab1, (SrGMV,SrCGMV,SrHRP,SrB),color=['m','c','y','g'])
ax.set_ylabel('Sharpe Ratio')
ax.set_xlabel('Type of allocation')
ax.set_title('Sharpe Ratios in-sample', pad=15)
ax.bar_label(p3,fmt='%.3f')
#PRE-COVID and POST-COVID
#evaluate mean and covariance with rolling method
#get prices of the last day of the month
d= pd.to_datetime(p.index)
pM=p.resample('M').last()
#compute monthly returns
rM= pM.pct_change(1)
#size of the window for the evaluation of mean and variance
w=60
#rolling Mean
rollM=rM.rolling(w).mean()
rollMean=rollM.dropna(0)
#rolling variance
rollVar=np.zeros((((rM.shape[0])-w),((rM.shape[1])),((rM.shape[1]))))
for j in range (w,((rM.shape[0]))):
    rollVar[(j-w),:,:]=rM.iloc[(j-w):j,0:rM.shape[1]].cov()
plt.figure()
((rollMean).plot()).yaxis.set_major_formatter(mtick.PercentFormatter(1.0))
plt.title('Rolling Average Monthly Returns', pad=15)
plt.legend(bbox_to_anchor=(1.05,0.2),loc="lower left",prop={'size': 10})
plt.xlabel('Time')
plt.ylabel('Returns in %')
#Split the sample in 2 time windows:
#1)from 2011 to February 2020 (Pre-Covid)
#2)from February 2020 till Decembre 2021 (Post-Covid)
#Computation of return and variance of each subperiod
#from 2011 to February 2020 (Pre-Covid)
rM1=r.loc[:"2020-02-24"]
mu1= rM1.mean()
mu1=((mu1.add(1)).pow(252)).sub(1)
S1 = rM1.cov()
S1=((S1.add(1)).pow(252)).sub(1)
#from February 2020 till today (Covid)
rM2=r.loc["2020-02-24":]
mu2= rM2.mean()
mu2=((mu2.add(1)).pow(252)).sub(1)
S2 = rM2.cov()
S2=((S2.add(1)).pow(252)).sub(1)
#compare returns
lab1=['World','EU-FR','EU-DE','EU-IT','EU-UK','America-US','America-CND','Oceania-
AUS','Africa-ZA','Asia-JPN','Asia-VN']
x = np.arange(len(lab1)) # the label locations
width = 0.35 # the width of the bars
fig, ax = plt.subplots()
ax.bar((x - width), mu1,width, label='Pre-Covid')
ax.bar((x),mu2,width,label='Post-Covid')
ax.set_ylabel('Return in %')
ax.set_title('Annualized historical average of daily returns', pad=15)

```

```

ax.yaxis.set_major_formatter(mtick.PercentFormatter(1.0,decimals=0))
plt.xticks(ticks=x, labels=lab1)
plt.xticks(rotation='vertical')
ax.legend()
plt.show()
#compare volatilities
plt.figure()
x = np.arange(len(lab1)) # the label locations
width = 0.35 # the width of the bars
fig, ax = plt.subplots()
ax.bar((x - width), (np.sqrt(np.diag(S1))),width, label='Pre-Covid',color='m')
ax.bar(x, (np.sqrt(np.diag(S2))),width,label='Post-Covid',color='y')
ax.set_ylabel('Volatility in %')
ax.set_title('Annualized historical volatilities of daily returns', pad=15)
ax.yaxis.set_major_formatter(mtick.PercentFormatter(1.0,decimals=0))
plt.xticks(ticks=x, labels=lab1)
plt.xticks(rotation='vertical')
ax.legend()
plt.show()
#subset 1
mu=mu1
S=S1
# Optimize for Minimum Volatility
ef = EfficientFrontier(mu[1:], S.iloc[1:,1:])
raw_weights_min=ef.min_volatility()
cleaned_weights_min= ef.clean_weights()
RetGMV1,SGMV1,SrGMV1=ef.portfolio_performance(verbose=True)
#Optimize with constrained GMV
#Group Constraints
#map each ticker to a geographic area
sector_mapper={'EU-FR':'EU', 'EU-DE':'EU', 'EU-IT':'EU', 'EU-UK':'EU', 'America-
US':'America', 'America-CND':'AmericaCND', 'Oceania-AUS':'Oceania', 'Africa-
ZA':'Emerging', 'Asia-JPN':'Asia', 'Asia-VN':'Emerging'}
sector_lower = {"America":0.40}
sector_upper = {"Emerging": 0.25}
efC = EfficientFrontier(mu[1:], S.iloc[1:,1:],weight_bounds=(0.03, 1))
for sector in sector_upper:
    is_sector = [v == sector for k, v in sector_mapper.items()]
    # Sum of weights from that sector has to be less than the upper bound
    efC.add_constraint(lambda w:sum(w[is_sector]) <= sector_upper[sector])
for sector in sector_lower:
    is_sector = [v == sector for k, v in sector_mapper.items()]
    efC.add_constraint(lambda w:sum(w[is_sector]) >= sector_lower[sector])
pwgtGMV=efC.min_volatility()
RetCGMV1,SCGMV1,SrCGMV1=efC.portfolio_performance(verbose=True)
#Optimize with HRP
#Compute the weights
hrp = HRPopt((r.loc["2020-02-24"]).iloc[:,1:])
hrp.optimize()
hrp_weights = hrp.clean_weights()
RetHRP1,SHRP1,SrHRP1=hrp.portfolio_performance(verbose=True)
#Comparison of weights

```

```

lab1=['EU-FR', 'EU-DE', 'EU-IT', 'EU-UK', 'America-US', 'America-CND', 'Oceania-
AUS', 'Africa-ZA', 'Asia-JPN', 'Asia-VN']
x = np.arange(len(lab1)) # the label locations
width = 0.25 # the width of the bars
fig, ax = plt.subplots()
ax.bar(x - width/2, cleaned_weights_min.values(), width, label='GMV
weights', color='m')
ax.bar(x + width/2, pwgtGMV.values(), width, label='C-GMV weights', color='b')
ax.bar((x+width*1.5), hrp_weights.values(), width, label='HRP', color='y')
ax.set_ylabel('Weights')
ax.set_title('Comparison of weights in "Pre-Covid" period', pad=15)
ax.yaxis.set_major_formatter(mtick.PercentFormatter(1.0, decimals=0))
plt.xticks(ticks=x, labels=lab1)
plt.xticks(rotation='vertical')
ax.legend()
plt.show()
#Benchmark
RetB1=(mu[0])
print('Expected annual return:', RetB1)
SB1=(np.sqrt(np.diag(S1)[0]))
print('Annual Volatility:', SB1)
SrB1=(RetB1-rf)/SB1
print('Sharpe Ratio:', SrB1)
#subset2
mu=mu2
S=S2
# Optimize for Minimum Volatility
ef = EfficientFrontier(mu[1:], S.iloc[1:,1:])
raw_weights_min=ef.min_volatility()
cleaned_weights_min= ef.clean_weights()
RetGMV2, SGMV2, SrGMV2=ef.portfolio_performance(verbose=True)
#Optimize with constrained GMV
#Group Constraints
#map each ticker to a geographic area
sector_mapper={'EU-FR': 'EU', 'EU-DE': 'EU', 'EU-IT': 'EU', 'EU-UK': 'EU', 'America-
US': 'America', 'America-CND': 'AmericaCND', 'Oceania-AUS': 'Oceania', 'Africa-
ZA': 'Emerging', 'Asia-JPN': 'Asia', 'Asia-VN': 'Emerging'}
sector_lower = {"America": 0.40}
sector_upper = {"Emerging": 0.25}
efC = EfficientFrontier(mu[1:], S.iloc[1:,1:], weight_bounds=(0.03, 1))
for sector in sector_upper:
    is_sector = [v == sector for k, v in sector_mapper.items()]
    # Sum of weights from that sector has to be less than the upper bound
    efC.add_constraint(lambda w: sum(w[is_sector]) <= sector_upper[sector])
for sector in sector_lower:
    is_sector = [v == sector for k, v in sector_mapper.items()]
    efC.add_constraint(lambda w: sum(w[is_sector]) >= sector_lower[sector])
pwgtGMV=efC.min_volatility()
RetCGMV2, SCGMV2, SrCGMV2=efC.portfolio_performance(verbose=True)
#Optimize with HRP
#Compute the weights
hrp = HRPopt((r.loc["2020-02-24":]).iloc[:,1:])
hrp.optimize()

```

```

hrp_weights = hrp.clean_weights()
RetHRP2,SHRP2,SrHRP2=hrp.portfolio_performance(verbose=True)
#Comparison of weights
lab1=['EU-FR','EU-DE','EU-IT','EU-UK','America-US','America-CND','Oceania-
AUS','Africa-ZA','Asia-JPN','Asia-VN']
x = np.arange(len(lab1)) # the label locations
width = 0.25 # the width of the bars
fig, ax = plt.subplots()
ax.bar(x - width/2,cleaned_weights_min.values(), width, label='GMV
weights',color='m')
ax.bar(x + width/2,pwgtGMV.values(), width, label='C-GMV weights',color='b')
ax.bar((x+width*1.5),hrp_weights.values(),width,label='HRP',color='y')
ax.set_ylabel('Weights')
ax.set_title('Comparison of weights in "Post-Covid" period')
ax.yaxis.set_major_formatter(mtick.PercentFormatter(1.0,decimals=0))
plt.xticks(ticks=x, labels=lab1)
plt.xticks(rotation='vertical')
ax.legend()
plt.show()
#Benchmark
RetB2=(mu[0])
print('Expected annual return:',RetB2)
SB2=(np.sqrt(np.diag(S)[0]))
print('Annual Volatility:',SB2)
SrB2=(RetB2-rf)/SB2
print('Sharpe Ratio:',SrB2)
#performance comparison across subperiods
#Returns
lab2=['GMV','GMV-C','HRP','Benchmark']
x = np.arange(len(lab2)) # the label locations
width = 0.25 # the width of the bars
fig, ax = plt.subplots()
ax.bar(x - width/2,(RetGMV1, RetCGMV1,RetHRP1,RetB1), width, label='Pre-Covid')
ax.bar(x + width/2,(RetGMV2, RetCGMV2,RetHRP2,RetB2), width, label='Post-Covid')
ax.set_ylabel('Return in %')
ax.set_title('Expected Performance Return',pad=15)
ax.yaxis.set_major_formatter(mtick.PercentFormatter(1.0,decimals=0))
plt.xticks(ticks=x, labels=lab2)
ax.legend(bbox_to_anchor=(1.05,0.7),loc="lower left",prop={'size': 10})
plt.show()
#Volatilities
lab2=['GMV','GMV-C','HRP','Benchmark']
x = np.arange(len(lab2)) # the label locations
width = 0.25 # the width of the bars
fig, ax = plt.subplots()
ax.bar(x - width/2,(SGMV1, SCGMV1,SHRP1,SB1), width, label='Pre-Covid',color='m')
ax.bar(x + width/2,(SGMV2, SCGMV2,SHRP2,SB2), width, label='Post-Covid',color='y')
ax.set_ylabel('Volatility in %')
ax.set_title('Expected Performance Volatilities ',pad=15)
ax.yaxis.set_major_formatter(mtick.PercentFormatter(1.0,decimals=0))
plt.xticks(ticks=x, labels=lab2)
ax.legend(bbox_to_anchor=(1.05,0.7),loc="lower left",prop={'size': 10})
plt.show()

```

```

#Sharpe Ratio
lab2=['GMV', 'GMV-C', 'HRP', 'Benchmark']
x = np.arange(len(lab2)) # the label locations
width = 0.25 # the width of the bars
fig, ax = plt.subplots()
ax.bar(x - width/2, (SrGMV1, SrCGMV1, SrHRP1, SrB1), width, label='Pre-Covid',
color='r')
ax.bar(x + width/2, (SrGMV2, SrCGMV2, SrHRP2, SrB2), width, label='Post-
Covid', color='b')
ax.set_ylabel('Sharpe Ratio')
ax.set_title('Sharpe Ratio Performance', pad=15)
plt.xticks(ticks=x, labels=lab2)
ax.legend(bbox_to_anchor=(1.05, 0.7), loc="lower left", prop={'size': 10})
plt.show()
#OUT-OF-SAMPLE
#Track Records
#data of interest are in r daily returns
rI=r.iloc[:,1:] #returns of the assets
rB=r.iloc[:,0]#returns of the benchmark
row_days=len(rI)
col_assets=len(rI.columns)
#window size for estimation/allocation
w=1260
rebal=21 # 1 month
#evaluation of asset moments
#estimation of expected returns
#pre-allocation for expected return computation
ErS=np.zeros((row_days-w, col_assets), dtype=float) #expected returns given sample
estimator
for j in range (w, row_days):
    #sample estimator
    ErS[(j-w), :] = ((rI.iloc[(j-w):j+1, :].mean()).add(1)).pow(252).sub(1)
#estimation of covariances
#pre-allocation for covariance computation
EvS=np.zeros(((row_days-w), col_assets, col_assets), dtype=float) #sample estimator
for j in range (w, row_days):
    #sample estimator
    EvS[(j-w), :, :] = ((rI.iloc[(j-w):(j+1), :].cov()).add(1)).pow(252).sub(1)
#evaluation of portfolio weights and realized returns
#CASE 1-GMV
#loop for the evaluation of realized returns with sample estimators of both
expected returns and covariance
#pre-allocation for returns
PortRetS1=np.empty(((row_days-w)), dtype=float)
PortRetS1[:]=np.NaN
#pre-allocation for weights
PortWS1=np.empty(((row_days-w), col_assets), dtype=float)
PortWS1[:, :]=np.NaN
#pre-allocation for volatilities
Vol1=np.empty((row_days-w), dtype=float)
Vol1[:]=np.NaN
for j in range(w+1, row_days+1, rebal):
    #set moments

```



```

mu=ErS[(j-w-1),:]
S=np.squeeze(EvS[(j-w-1),:,:])
#GMV
ef = EfficientFrontier(mu,S)
pwgtGMV=np.array(list(ef.min_volatility().values()))
PortWS1[(j-w-1),:]=np.transpose(pwgtGMV)
PortRetS1[(j-w-1)]=np.dot(np.array(rI.iloc[(j-1),:]),(pwgtGMV))
Vol1[(j-w-1)]=np.sqrt(np.dot(np.dot(pwgtGMV,S), pwgtGMV))
#fill the weights, the returns and the volatilities
for j in range(w+1,row_days+1):
    PortWS1[(j-w-1),:]=np.nan_to_num(PortWS1[(j-w-1),:],nan=PortWS1[(j-w-1-1)])
    PortRetS1[(j-w-1)]=np.nan_to_num(PortRetS1[(j-w-
1)],nan=np.dot(np.array(rI.iloc[(j-1),:]),np.transpose(PortWS1[(j-w-1-1),:])))
    Vol1[(j-w-1)]=np.nan_to_num(Vol1[(j-w-
1)],nan=np.sqrt(np.dot(np.dot(np.transpose(PortWS1[(j-w-1-1),:]),
np.squeeze(EvS[(j-w-1),:,:])), np.transpose(PortWS1[(j-w-1-1),:])))
#Sharpe Ratio GMV out of sample
VolGMV=np.nanstd(PortRetS1)*np.sqrt(252)
RetGMVofs=((np.nanmean(PortRetS1)+1)**(252))-(1)
Sr1ofs=(RetGMVofs-rf)/VolGMV
#kernel density volatility GMV
d1=pd.DataFrame(Vol1-np.nanmean(Vol1))
d1.plot.kde(bw_method=True)
plt.title('KDE-Density plot for Volatility GMV')
plt.show()
#kernel density return GMV
d1r=pd.DataFrame(PortRetS1-np.nanmean(PortRetS1))
d1r.plot.kde(bw_method=True)
plt.title('KDE-Density plot for Returns GMV')
plt.show()
#Descriptive statistic for the GMV
#mean, median,standard deviation, skweness, kurtosis and interquartile range of
volatilites
m1GMV=float((pd.DataFrame(Vol1).mean(numeric_only=True)))
m2GMV=float((pd.DataFrame(Vol1).median(numeric_only=True)))
m3GMV=float((pd.DataFrame(Vol1).var(numeric_only=True)))
m4GMV=float(pd.DataFrame(Vol1).std(numeric_only=True))
m5GMV=float(pd.DataFrame(Vol1).skew(numeric_only=True))
m6GMV=float(pd.DataFrame(Vol1).kurtosis(numeric_only=True))
#mean, median, standard deviation, skweness , kurtosis and interquartile range of
returns
r1GMV=float((pd.DataFrame(PortRetS1).mean(numeric_only=True))+1)**(252)-(1)
r2GMV=float((pd.DataFrame(PortRetS1).median(numeric_only=True))+1)**(252)-(1)
r3GMV=float((pd.DataFrame(PortRetS1).var(numeric_only=True))+1)**(252)-(1)
r4GMV=float(pd.DataFrame(PortRetS1).std(numeric_only=True))*np.sqrt(252)
r5GMV=float(pd.DataFrame(PortRetS1).skew(numeric_only=True))
r6GMV=float(pd.DataFrame(PortRetS1).kurtosis(numeric_only=True))
#CASE 2-GMV constrained
#pre-allocation for returns
w1=1260
PortRetS2=np.empty(((row_days-w1)),dtype=float)
PortRetS2[:]=np.NaN
#pre-allocation for weights

```

```

PortWS2=np.empty(((row_days-w1),col_assets),dtype=float)
PortWS2[:,:]=np.NaN
#pre-allocation for volatilities
Vol2=np.empty(((row_days-w1)),dtype=float)
Vol2[:,:]=np.NaN
#creating portfolio
for j in range(w1+1,row_days,reb):
    #set moments
    mu=ErS[(j-w1-1),:]
    S=np.squeeze(EvS[(j-w1-1),:,:])
    #GMV
    efC=EfficientFrontier(mu,S,weight_bounds=(0.03, 1))
    for sector in sector_upper:
        is_sector = [v == sector for k, v in sector_mapper.items()]
        # Sum of weights from that sector has to be less than the upper bound
        efC.add_constraint(lambda w:sum(w[is_sector]) <= sector_upper[sector])
    for sector in sector_lower:
        is_sector = [v == sector for k, v in sector_mapper.items()]
        efC.add_constraint(lambda w:sum(w[is_sector]) >= sector_lower[sector])
    pwgtGMV=np.array(list(efC.min_volatility().values()))
    PortWS2[(j-w1-1),:]=np.transpose(pwgtGMV)
    PortRetS2[(j-w1)]=np.dot(np.array(rI.iloc[(j-1),:]),(pwgtGMV))
    Vol2[(j-w1)]=np.sqrt(np.dot(np.dot(pwgtGMV,S), pwgtGMV))
#fill the weights, the returns and the volatilities
for j in range(w+1,row_days+1):
    PortWS2[(j-w-1),:]=np.nan_to_num(PortWS2[(j-w-1),:],nan=PortWS2[(j-w-1-1)])
    PortRetS2[(j-w-1)]=np.nan_to_num(PortRetS2[(j-w-1-1)],nan=np.dot(np.array(rI.iloc[(j-1),:]),np.transpose(PortWS2[(j-w-1-1),:])))
    Vol2[(j-w-1)]=np.nan_to_num(Vol2[(j-w-1-1)],nan=np.sqrt(np.dot(np.dot(np.transpose(PortWS2[(j-w-1-1),:]),np.squeeze(EvS[(j-w-1),:,:])), np.transpose(PortWS2[(j-w-1-1),:])))
#Sharpe Ratio GMV constrained out of sample
VolGMVC=np.nanstd(PortRetS2)*np.sqrt(252)
RetCGMVofs=(np.nanmean(PortRetS2)+1)**(252)-(1)
Sr2ofs=(RetCGMVofs-rf)/VolGMVC
#kernel density volatility GMV constrained
d2=pd.DataFrame(Vol2-np.nanmean(Vol2))
d2.plot.kde(bw_method=True)
plt.title('KDE-Density plot for Volatility GMV constrained')
plt.show()
#kernel density returns GMV constrained
r2=pd.DataFrame(PortRetS2-np.nanmean(PortRetS2))
r2.plot.kde(bw_method=True)
plt.title('KDE-Density plot for Returns GMV constrained')
plt.show()
#Descriptive statistic for the GMV constrained
#mean, median,standard deviation, skweness, kurtosis and interquartile range of
volatilites
m1GMVC=(float(pd.DataFrame(Vol2).mean(numeric_only=True)))
m2GMVC=(float(pd.DataFrame(Vol2).median(numeric_only=True)))
m3GMVC=float((pd.DataFrame(Vol2).var(numeric_only=True)))
m4GMVC=float(pd.DataFrame(Vol2).std(numeric_only=True))
m5GMVC=float(pd.DataFrame(Vol2).skew(numeric_only=True))

```

```

m6GMVC=float(pd.DataFrame(Vol2).kurtosis(numeric_only=True))
#mean, median, standard deviation, skweness , kurtosis and interquartile range of
returns
r1GMVC=(float(pd.DataFrame(PortRetS2).mean(numeric_only=True))+1)**(252)-(1)
r2GMVC=(float(pd.DataFrame(PortRetS2).median(numeric_only=True))+1)**(252)-(1)
r3GMVC=float((pd.DataFrame(PortRetS2).var(numeric_only=True))+1)**(252)-(1)
r4GMVC=float(pd.DataFrame(PortRetS2).std(numeric_only=True))*np.sqrt(252)
r5GMVC=float(pd.DataFrame(PortRetS2).skew(numeric_only=True))
r6GMVC=float(pd.DataFrame(PortRetS2).kurtosis(numeric_only=True))
#CASE 3-HRP
#pre-allocation for returns
PortRetHRP=np.empty(((row_days-w)),dtype=float)
PortRetHRP[:]=np.NaN
#pre-allocation for weights
PortWHRP=np.empty(((row_days-w),col_assets),dtype=float)
PortWHRP[:,:]=np.NaN
#pre-allocation for volatilities
Vol3=np.empty((row_days-w),dtype=float)
Vol3[:]=np.NaN
#creating portfolio
for j in range(w+1,row_days+1,rebals):
    #set the inputs
    rets=rI.iloc[(j-w-1):j,:]
    hrp = HRPopt(rets)
    #HRP
    pwtgHRP = np.array(list(hrp.optimize().values()))
    PortWHRP[(j-w-1),:]=np.transpose(pwtgHRP)
    PortRetHRP[(j-w-1)]=np.dot(np.array(rI.iloc[(j-1),:]),(pwtgHRP))
    S=np.squeeze(EvS[(j-w-1),:,:])
    Vol3[(j-w-1)]=np.sqrt(np.dot(np.dot(pwtgHRP,S), pwtgHRP))
#fill the weights, the returns and the volatilities
for j in range(w+1,row_days+1):
    PortWHRP[(j-w-1),:]=np.nan_to_num(PortWHRP[(j-w-1),:],nan=PortWHRP[(j-w-1-1)])
    PortRetHRP[(j-w-1)]=np.nan_to_num(PortRetHRP[(j-w-1-1)],nan=np.dot(np.array(rI.iloc[(j-1),:]),np.transpose(PortWHRP[(j-w-1-1),:])))
    Vol3[(j-w-1)]=np.nan_to_num(Vol3[(j-w-1-1)],nan=np.sqrt(np.dot(np.dot(np.transpose(PortWHRP[(j-w-1-1),:]),
np.squeeze(EvS[(j-w-1),:,:]), np.transpose(PortWHRP[(j-w-1-1),:]))))
#Sharpe Ratio HRP out of sample
VolHRP=np.nanstd(PortRetHRP)*np.sqrt(252)
RetHRPofs=((np.nanmean(PortRetHRP))+1)**(252)-(1)
Sr3ofs=(RetHRPofs-rf)/VolHRP
#kernel density volatility HRP
pd.DataFrame(Vol3-np.nanmean(Vol3)).plot.kde(bw_method=True)
plt.title('KDE-Density plot for Volatility HRP')
plt.show()
#kernel density returns HRP
r3=pd.DataFrame(PortRetHRP-np.nanmean(PortRetHRP))
r3.plot.kde(bw_method=True)
plt.title('KDE-Density plot for Returns HRP')
plt.show()
#Descriptive statistic for the HRP
#mean, median,standard deviation, skweness and kurtosis of volatilities

```

```

m1HRP=(float(pd.DataFrame(Vol3).mean(numeric_only=True)))
m2HRP=(float(pd.DataFrame(Vol3).median(numeric_only=True)))
m3HRP=float((pd.DataFrame(Vol3).var(numeric_only=True)))
m4HRP=float(pd.DataFrame(Vol3).std(numeric_only=True))
m5HRP=float(pd.DataFrame(Vol3).skew(numeric_only=True))
m6HRP=float(pd.DataFrame(Vol3).kurtosis(numeric_only=True))
#mean, median, standard deviation, skwenessa and kurtosis of returns
r1HRP=(float(pd.DataFrame(PortRetHRP).mean(numeric_only=True))+1)**(252)-(1)
r2HRP=(float(pd.DataFrame(PortRetHRP).median(numeric_only=True))+1)**(252)-(1)
r3HRP=float((pd.DataFrame(PortRetHRP).var(numeric_only=True))+1)**(252)-(1)
r4HRP=float(pd.DataFrame(PortRetHRP).std(numeric_only=True))*np.sqrt(252)
r5HRP=float(pd.DataFrame(PortRetHRP).skew(numeric_only=True))
r6HRP=float(pd.DataFrame(PortRetHRP).kurtosis(numeric_only=True))
#area plot for weights over time
#plots the weights for each portoflio
#GMV
plt.figure()
pd.DataFrame(np.squeeze(PortWS1[:, :])).plot.area()
plt.legend(loc='center left', bbox_to_anchor=(1.5, 0.5))
plt.title("Evolution of weights overtime with GMV portfolio", pad=15)
plt.ylabel('Weights sum to 1')
#GMV constrained
plt.figure()
pd.DataFrame(np.squeeze(PortWS2[:, :])).plot.area()
plt.title("Evolution of weights overtime with Constrained GMV portfolio", pad=15)
plt.legend(loc='center left', bbox_to_anchor=(1.5, 0.5))
plt.ylabel('Weights sum to 1')
#HRP
plt.figure()
pd.DataFrame(PortWHRP).plot.area()
plt.title("Evolution of weights overtime with HRP portfolio", pad=15)
plt.legend(loc='center left', bbox_to_anchor=(1.5, 0.5))
plt.ylabel('Weights sum to 1')
#Approximate Turnover
#GMV
w=pd.DataFrame(np.squeeze(PortWS1[:, :]))
tS1GMV=w.subtract(w.shift(periods=-1, axis=0, fill_value=0))
tS1GMV=(pd.DataFrame(np.dot(abs(tS1GMV),np.ones((len(tS1GMV.columns),1),dtype=float
))))).mul(0.5)
tS1GMV=(tS1GMV.loc[(tS1GMV != 0).any(axis=1),:][::-1]
#GMV constrained
w=pd.DataFrame(np.squeeze(PortWS2[:, :]))
tS2GMV=w.subtract(w.shift(periods=-1, axis=0, fill_value=0))
tS2GMV=(pd.DataFrame(np.dot(abs(tS2GMV),np.ones((len(tS2GMV.columns),1),dtype=float
))))).mul(0.5)
tS2GMV=(tS2GMV.loc[(tS2GMV != 0).any(axis=1),:][::-1]
#HRP
w=pd.DataFrame(np.squeeze(PortWHRP[:, :]))
tS1HRP=w.subtract(w.shift(periods=-1, axis=0, fill_value=0))
tS1HRP=(pd.DataFrame(np.dot(abs(tS1HRP),np.ones((len(tS1HRP.columns),1),dtype=float
))))).mul(0.5)
tS1HRP=(tS1HRP.loc[(tS1HRP != 0).any(axis=1),:][::-1]
#Plot

```

```

fig=plt.figure()
ax = plt.subplot(1,1,1)
p3=ax.plot(tS1GMV)
p4=ax.plot(tS2GMV)
p5=ax.plot(tS1HRP)
ax.yaxis.set_major_formatter(mtick.PercentFormatter(1.0,decimals=0))
ax.set_ylabel('Change in weights in % ')
ax.set_xlabel('Rebalance date',labelpad=30)
ax.set_title('Turnover Rate of allocations with monthly rebalacing', pad=15)
plt.legend(['GMV', 'C-GMV', 'HRP'])
plt.show()
#Max turnover
print(tS1GMV.max())
print(tS2GMV.max())
print(tS1HRP.max())
#Monitoring diversification
w1=pd.DataFrame(np.squeeze(PortWS1[:,:])) #GMV portfolio with rolling sample
moments
wEW= np.dot((1/15),np.ones((len(w1),len(w1.columns)),dtype=float))
d1=abs(w1-wEW).sum(1)
plt.figure()
plt.plot(d1,color='g')
w2=pd.DataFrame(np.squeeze(PortWS2[:,:])) #GMV constrained portfolio with rolling
sample moments
d2=abs(w2-wEW).sum(1)
plt.plot(d2,color='b')
w3=pd.DataFrame(np.squeeze(PortWHRP[:,:])) #HRP
d3=abs(w3-wEW).sum(1)
plt.plot(d3,color='r')
plt.legend(['GMV', 'GMV constrained', 'HRP'])
plt.title('Diversification Index')
w=1260
#cumulated returns
#cumulated returns of the indices
CRrI=((rI.iloc[w:,:].div(100))+1).cumprod()
#cumulated returns of the market
CRrB=((rB.iloc[w:].div(100))+1).cumprod()
#cumulated returns of the strategies
CRS1GMV=((pd.DataFrame(PortRetS1).iloc[:,0].div(100))+1).cumprod()
CRS2GMV=((pd.DataFrame(PortRetS2).iloc[:,0].div(100))+1).cumprod()
CR3HRP=((pd.DataFrame(PortRetHRP).iloc[:,0].div(100))+1).cumprod()
#plot the cumulated returns
#market
plt.figure()
plt.plot(CRrB)
plt.legend(['World'])
plt.title('Cumulated Returns')
#indices
plt.figure()
plt.plot(CRrI)
plt.legend(lab[1:])
plt.title('Cumulated Returns')
#GMV and HRP baseline

```

```

plt.figure()
plt.plot(CRS1GMV)
plt.plot(CR3HRP)
plt.legend(['GMV sample baseline', 'HRP'])
plt.title('Cumulated Returns')
#GMV across constraints and market
plt.figure()
plt.plot(CRS1GMV)
plt.plot(CRS2GMV)
#plt.plot(CRrB)
plt.legend(['GMV sample baseline', 'GMV constraints'])
plt.title('Cumulated Returns')
#Comparison by means of performance measures
#Returns out of sample
lab2=['GMV', 'GMV-C', 'HRP']
fig=plt.figure()
ax=plt.subplot(1,1,1)
p4=plt.bar(lab2, (RetGMVofs, RetCGMVofs, RetHRPofs), color=['m', 'c', 'y'])
ax.yaxis.set_major_formatter(mtick.PercentFormatter(1.0, decimals=0))
ax.set_ylabel('Return in %')
ax.set_xlabel('Type of allocation')
ax.bar_label(p4, fmt='%.3f')
ax.set_title('Expected Annualized Performance Returns out-of-sample', pad=15)
plt.show()
print ((RetGMVofs, RetCGMVofs, RetHRPofs))
#Volatilities out of sample
fig=plt.figure()
ax=plt.subplot(1,1,1)
p5=plt.bar(lab2, (VolGMV, VolGMVC, VolHRP), color=['m', 'c', 'y'])
ax.yaxis.set_major_formatter(mtick.PercentFormatter(1.0, decimals=0))
ax.set_ylabel('Return in %')
ax.set_xlabel('Type of allocation')
ax.bar_label(p5, fmt='%.5f')
ax.set_title('Expected Annualized Performance Volatilities out-of-sample', pad=15)
plt.show()
print (VolGMV, VolGMVC, VolHRP)
#Sharpe Ratios out of sample
fig=plt.figure()
ax = plt.subplot(1,1,1)
p6=plt.bar(lab2, (Sr1ofs, Sr2ofs, Sr3ofs), color=['m', 'c', 'y'])
ax.set_ylabel('Sharpe Ratio')
ax.set_xlabel('Type of allocation')
ax.set_title('Sharpe Ratios out-of-sample', pad=15)
ax.bar_label(p6, fmt='%.3f')
plt.show()
print (Sr1ofs, Sr2ofs, Sr3ofs)
#sharpe ratios in sample
fig=plt.figure()
ax = plt.subplot(1,1,1)
p7=ax.bar(lab2, (SrGMV, SrCGMV, SrHRP), color=['m', 'c', 'y'])
ax.set_ylabel('Sharpe Ratio')
ax.set_xlabel('Type of allocation')
ax.set_title('Sharpe Ratios in-sample', pad=15)

```

```

ax.bar_label(p7,fmt='%.3f')
#Comparison in vs out-of-sample for GMV, GMV constrained, HRP
#Returns
lab2=['GMV','GMV-C','HRP']
x = np.arange(len(lab2)) # the label locations
width = 0.25 # the width of the bars
fig, ax = plt.subplots()
ax.bar(x - width/2,(RetGMV, RetCGMV,RetHRP), width, label='In sample', color='y')
ax.bar(x + width/2,(RetGMVofs, RetCGMVs,RetHRPofs), width, label='Out of sample')
ax.set_ylabel('Return')
ax.set_title('Comparison of returns')
plt.xticks(ticks=x, labels=lab2)
ax.legend()
plt.show()
#Volatilities
lab2=['GMV','GMV-C','HRP']
x = np.arange(len(lab2)) # the label locations
width = 0.25 # the width of the bars
fig, ax = plt.subplots()
ax.bar(x - width/2,(SGMV, SCGMV,SHRP), width, label='In sample',color='m')
ax.bar(x + width/2,(VolGMV, VolGMVC,VolHRP), width, label='Out of sample')
ax.set_ylabel('Volatility')
ax.set_title('Comparison of Volatilities')
plt.xticks(ticks=x, labels=lab2)
ax.legend()
plt.show()
#Sharpe Ratio
lab2=['GMV','GMV-C','HRP']
x = np.arange(len(lab2)) # the label locations
width = 0.25 # the width of the bars
fig, ax = plt.subplots()
ax.bar(x - width/2,(SrGMV, SrCGMV,SrHRP), width, label='In sample')
ax.bar(x + width/2,(Sr1ofs, Sr2ofs,Sr3ofs), width, label='Out of sample')
ax.set_ylabel('Sharpe Ratio')
ax.set_title('Comparison of Sharpe Ratios')
plt.xticks(ticks=x, labels=lab2)
ax.legend()
plt.show()
#plot density volatilities
plt.figure()
pd.DataFrame({'GMV':Vol1-np.nanmean(Vol1),'Constrained GMV':Vol2-
np.nanmean(Vol2),'HRP':Vol3-np.nanmean(Vol3)}).plot.kde(bw_method=True)
plt.legend(['GMV','Constrained GMV','HRP'])
plt.title('KDE-Density plot for Volatility')
plt.show()
#Descriptive statistics of volatilities
measures = {'Mean': [m1GMV, m1GMVC, m1HRP],'Median':
[m2GMV,m2GMVC,m2HRP],'Variance': [m3GMV,m3GMVC,m3HRP],'Standard deviation':
[m4GMV,m4GMVC,m4HRP],'Skewness': [m5GMV,m5GMVC,m5HRP], 'Kurtosis':
[m6GMV,m6GMVC,m6HRP]}#,'Interquartile Range': [m7GMV,m7GMVC,m7HRP]}
Tab = pd.DataFrame(measures)
Tab.index=['GMV sample baseline','GMV sample constraints','HRP']
Tab.style

```

```

#Descriptive statistics of returns
measures1 = {'Mean': [r1GMV, r1GMVC, r1HRP], 'Median':
[r2GMV, r2GMVC, r2HRP], 'Variance': [r3GMV, r3GMVC, r3HRP], 'Standard deviation':
[r4GMV, r4GMVC, r4HRP], 'Skewness': [r5GMV, r5GMVC, r5HRP], 'Kurtosis':
[r6GMV, r6GMVC, r6HRP]}#, 'Interquartile Range': [r7GMV, r7GMVC, r7HRP]}
Tab1 = pd.DataFrame(measures1)
Tab1.index=['GMV sample baseline', 'GMV sample constraints', 'HRP']
Tab1.style
#plot density returns
plt.figure()
pd.DataFrame({'GMV':PortRetS1-np.nanmean(PortRetS1), 'Constrained GMV':PortRetS2-
np.nanmean(PortRetS2), 'HRP':PortRetHRP-
np.nanmean(PortRetHRP)}).plot.kde(bw_method=True)
plt.legend(['GMV', 'Constrained GMV', 'HRP'])
plt.title('KDE-Density plot for returns')
plt.show()

```


REFERENCES

1. ALLEN, G. C., 2010. The Risk Parity Approach to Asset Allocation. *Callan Investments Institute Research* [online]. Available at <<https://web.archive.org/web/20110930074457/http://www.callan.com/research/download/?file=papers%2Ffree%2F368.pdf>> [Date of access: 06/01/2021].
2. ARAGÓN, A., et al., 2017. Clustering algorithms for Risk-Adjusted Portfolio Construction. *Procedia Computer Science* [online], 108, pp. 1334-1343. Available at <<https://www.sciencedirect.com/science/article/pii/S187705091730772X>> [Date of access: 10/10/2021].
3. BAILEY, D. H., and DE PRADO, M. L., 2012. Balanced Basket: A new approach to Trading and Hedging Risks. *Journal of Investment Strategies* [online], 1 (4), pp. 21-62. Available at <https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2066170> [Date of access: 15/01/2021].
4. BAILEY, D. H., and DE PRADO, M. L., 2013. An Open-Source Implementation of the Critical-Line Algorithm for Portfolio Optimization. *Algorithms* [online], 6 (1), pp. 169-196. Available at <https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2197616> [Date of access: 15/10/2021].
5. BAILEY, D. H., and DE PRADO, M. L., 2014. Pseudo-Mathematics and Financial Charlatanism: The Effects of Backtest Overfitting on Out-Of-Sample Performance. *Notice of the American Mathematical Society* [online], 61 (5), pp. 458-471. Available at <https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2308659> [Date of access: 12/01/2021].
6. BARUCCI, E., and FONTANA, C., 2017. *Financial markets theory. Equilibrium, efficiency and information*. 2° ed. London: Springer.
7. BELSEY, D. A., KUH, E. and WELSCH, R. E., 2004. *Regression diagnostic. Identifying Influential Data and Sources of Collinearity*. 1°ed. Hoboken: Wiley-Interscience.
8. CHOUEIFATY, Y. and COIGNARD, Y., 2008. Toward Maximum Diversification. *Journal of Portfolio Management* [online], 35 (1), pp. 40-51. Available at <<https://www.tobam.fr/wp-content/uploads/2014/12/TOBAM-JoPM-Maximum-Div-2008.pdf>> [Date of access: 06/01/2022].
9. DALIO, R., JENSEN, G., and DALIO, R., 2015. Our Thoughts about Risk Parity and All Weather. *Bridgewater Daily Observations* [online]. Available at

- <http://2zfrwd1id94z1yvpxlk5mcn-wpengine.netdna-ssl.com/wp-content/uploads/2015/10/Our-Thoughts-about-Risk-Parity-and-All-Weather-Bridgewater-Ray-Dalio-2015.pdf>> [Date of access: 15/01/202].
10. Directive of the European Parliament 2014/65/EU of the 30th of April 2014. Markets in Financial Instruments Directive (MiFID II) [online]. Available at <<https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32014L0065&from=IT>> [Date of access: 18/01/2022].
 11. HURST, B., JOHNSON, B., and OOI, Y.H., 2010. *Understanding Risk Parity*. AQR Capital Management, LLC [online]. Available at <<https://www.aqr.com/-/media/AQR/Documents/Insights/White-Papers/Understanding-Risk-Parity.pdf>> [Date of access: 13/11/2021].
 12. JAEGER, M., et al., 2021. Interpretable Machine Learning for Diversified Portfolio Construction. *The Journal of Financial Data Science* [online]. Available at <https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3730144> [Date of access: 10/10/2021].
 13. JAIN, P., and JAIN, S., 2019. Can Machine Learning-Based Portfolios Outperform Traditional Risk-Based Portfolios? The Need to Account for Covariance Misspecification. *Risks* [online], 7 (3). Available at <<https://www.mdpi.com/2227-9091/7/3/74>> [Date of access: 15/10/2021].
 14. JOHNSON, S., 2022. Net growth in ETF numbers almost doubles to a record 1,239 last year. *Financial Times* [online]. Available at <<https://www.ft.com/content/85dd6319-4a24-41bb-b7fa-60e67eeb7ef4#comments-anchor>> [Date of access: 18/01/2022].
 15. JURCZENKO, E., 2015. *Risk-Based and Factor Investing*. 1^o. Oxford: Elsevier.
 16. KATZKE, N., and PFITZINGER, J., 2019. A constrained hierarchical risk parity algorithm with cluster-based capital allocation. *Working Papers* [online]. Stellenbosch University, Departments of Economics. Available at <<https://ideas.repec.org/p/sza/wpaper/wpapers328.html>> [Date of access: 15/10/2021].
 17. KLEIN, M., C., 2015. *Financial Times* [online]. Available at <<https://www.ft.com/content/23274f79-2545-376c-b2da-63f9332a2e71>> [Date of access: 16/01/2022].
 18. LOHRE, H., ROTHER, C., and SCHÄFER, K.A., 2020. Hierarchical Risk Parity: Accounting for Tail Dependencies in Multi-Asset Multi-Factor Allocations. In JURCZENKO, E., 2020. *Machine Learning and Asset Management* [online]. 1^oed. Iste and Wiley. [online]. Available at

- <https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3513399> [Date of access: 12/10/2021].
19. LÓPEZ DE PRADO, M., 2016. Building Diversified Portfolios that Outperform Out-of-Sample. *Journal of Portfolio Management* [online], 42 (4), pp. 59-69. Available at <https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2708678> [Date of access: 15/10/2021].
 20. LÓPEZ DE PRADO, M., 2018. *Advances in Financial Machine Learning*. 1° ed. Hoboken: Wiley and Sons.
 21. MANGRAM, M. E., 2013. A simplified perspective of the Markowitz portfolio theory. *Global Journal of Business Research*, 7 (1), pp. 59-70. Available at <https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2147880> [Date of access: 09/11/2021].
 22. MARKOWITZ, H., 1952. Portfolio Selection. *The Journal of Finance* [online], 7 (1), pp.77-91. Available at <https://www.math.hkust.edu.hk/~maykwok/courses/ma362/07F/markowitz_JF.pdf> [Date access: 08/11/2021].
 23. MARKOWITZ, H., and TODD, G.P., 2000. *Mean variance analysis in portfolio choice and capital markets*. 1° ed. New York: John Wiley and Sons.
 24. MARTIN, R. A., 2021. PyPortfolioOpt: portfolio optimization in Python. *Journal of Open Source Software* [online], 6 (61). Available at <<https://doi.org/10.21105/joss.03066>> [Date of access: 12/11/2021].
 25. RAFFINOT, T., 2018. *The Hierarchical Equal Risk Contribution Portfolio* [online]. AXA Investment Managers. Available at <https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3237540> [Date of access: 13/10/2021].
 26. RAYCHAUDHURI, S., 2008. Introduction to Monte Carlo simulation. *2008 Winter Simulation Conference* [online], pp. 91-100. Available at <<https://www.informs-sim.org/wsc08papers/012.pdf>> [Date of access: 16/01/2021].
 27. SECURITIES AND EXCHANGE COMMISSION: OFFICE OF INVESTOR EDUCATION AND ADVOCAY, 2012. Exchange-Traded Funds (ETFs). *Investor Bulletin* [online]. Available at: <<https://www.sec.gov/investor/alerts/etfs.pdf>> [Date of access: 25/12/2021].
 28. SIEGEL, J., J., 2014. *Stocks for the Long Run: The Definitive Guide to Financial Market Returns and Long-Term Investment Strategies*. 5° ed. New York: Mc Graw Hill Education.

SITOGRAPHY

1. <<https://www.borsaitaliana.it/etf/formazione/cosaeunetf/coseunetf.htm>> [Date of access: 23/12/2021].
2. <<https://corporatefinanceinstitute.com/resources/knowledge/trading-investing/efficient-frontier/>> [Date of access: 28/12/2021].
3. <https://www.ecb.europa.eu/pub/economic-bulletin/focus/2020/html/ecb.ebbox202005_03~12b5ff68bf.en.html> [Date of access: 16/01/2022].
4. <<https://etfgi.com/news/press-releases/2021/08/etfgi-reports-assets-global-etfs-industry-reach-record-us946-trillion>> [Date of access: 18/01/2022].
5. <<https://www.ibm.com/cloud/learn/unsupervised-learning>> [Date of access: 09/11/2021].
6. <<https://www.investor.gov/introduction-investing/investing-basics/investment-products/mutual-funds-and-exchange-traded-2>> [Date of access: 26/12/2021].
7. <<https://www.linkedin.com/in/lopezdeprado/>> [Date of access: 12/10/2021].
8. <<https://www.morningstar.com/articles/1071491/the-year-in-etfs>> [Date of access: 18/01/2022].
9. <<https://www.msci.com/documents/10199/149ed7bc-316e-4b4c-8ea4-43fcb5bd6523>> [Date of access: 26/12/2021].
10. <<https://www.nasdaq.com/glossary/p/principal-of-diversification>> [Date of access: 10/12/2021].
11. <<https://pyportfolioopt.readthedocs.io/en/latest/index.html>> [Date of access: 12/11/2021].
12. <<https://www.quantresearch.org>> [Date of access: 12/10/2021]

