BI Norwegian Business School - campus Oslo

GRA 19502

Master Thesis

Component of continuous assessment: Thesis Master of Science
Final master thesis – Counts 80% of total grade

Asset allocation of the Norwegian Government Pension Fund Global with programming in Python

Navn:	Ida Christine Korme Thorp, Martine Hauahei Yttervik
Start:	02.03.2018 09.00
Finish:	03.09.2018 12.00

Martine Hauahei Yttervik

Ida Christine Korme Thorp

Hand-in date:

03.09.2018

Programme:
Master of Science in Business, Major in Finance

Master of Science in Business, Major in Economics

Acknowledgements

Our motivation for writing this master thesis within the field of quantitative and qualitative finance, origins in various courses taught in financial theory throughout our education at BI Norwegian Business School. This opened our eyes for mathematical finance, and when searching for a thesis topic it was natural for us to return to an area we knew would be both interesting and challenging. With our different backgrounds, in economics and finance, we felt this topic could benefit from both faculties. Working with this thesis has given us a deeper understanding in the extended field of finance, as well as highly useful skills, such as programming in Python.

We would like to express our deepest gratitude to our supervisor, Professor Alfonso Irarrazabal, for his contribution and guidance throughout our research. When the opportunity to write under the inspiration and supervision of Professor Irarrazabal arose, it was an obvious choice for us given his hands-on experience within the topic of our interest. Last but not least, we would also like to thank our closest family and friends for the support and help.

Abstract

This thesis aims to assess if the optimal asset allocation for the Norwegian Government Pension Fund – Global could be improved. We were curious to see if we were able to optimize the portfolio by only looking at the risk-return relationship, without taking political, economic or ethical interest into evaluation. Since Norges Bank Investment Management has expressed that they do not have the absolute answer for what is the optimal asset allocation, we were interested to research whether or not we could obtain better results by purely examine the financial performance. The research set out to calculate the optimal portfolio weighting from historical data collected from 2008 until 2018. To analyze and compare the results we used the FTSE Benchmark Index, which is the benchmark used for the Government Pension Fund – Global. Therefore, we tried to replicate the FTSE benchmark by using 25 of the same countries in our portfolio. Upon advice from our supervisor we chose to program our own portfolio optimizer from scratch with Python as our programming tool. We constructed the different portfolios and divided them into constant expected return and time-varying expected return. Even though this was much more time consuming than using another software, we found it to be rewarding. As expected, our results showed that with our asset allocation we did not outperform the benchmark, except one portfolio that is rather close. However, this portfolio was closer than one should assume – compared to the benchmark Norges Bank uses that has considerably more complexity and more political-economic decisions behind its investment strategy. So – is it possible to rather concentrate on the return and variance trade-off instead off introducing the vast complexity of several influencing factors.

Table of Content

List of figures

List of tables

1. Introduction

One of the most important objectives in asset management is to make decisions concerning the optimal asset allocation (Sharpe W. , 1964). The goal of asset managers is to realize the highest possible return corrected for risk, with the lowest value at risk (VaR) possible, and in response to news, prices of different asset classes changes in anticipation of future performance. When it comes to portfolio choices and in determining aggregate risk, the structure of variance and correlation across assets are extremely important. The goal of this thesis is to explore what is the best possible asset allocation for the Norwegian Government Pension Fund - Global. We will compare the FTSE Benchmark they are using today and try to optimize the portfolio by using asset allocation to change the weights they are investing in per country.

The Norwegian Government Pension Fund - Global was founded in 1990 by The Norwegian Government as a fiscal policy tool to manage and preserve the rapid growing petroleum revenues into long-term investments. The purpose of the Fund is to benefit and save for future generations in Norway, as well as giving the Norwegian government a tool to stabilize and stimulate the economy. Further in this thesis we will refer to the Norwegian Government Pension Fund - Global as the Fund or the abbreviation GPFG, and Norges Bank Investment Management as the abbreviation NBIM. Today all the Norwegian government's oil and gas revenues are transferred directly into the fund and invested into three categories which are divided as follows; 66,8 % in equity-, 31,6% in fixed income -, and 2,5% in unlisted real estate investments. The market value for the Fund has increased steady over the last decade, making it one of the world's largest funds with a current market value of approximately NOK 8.500 billion as you can see in figure 1 below.

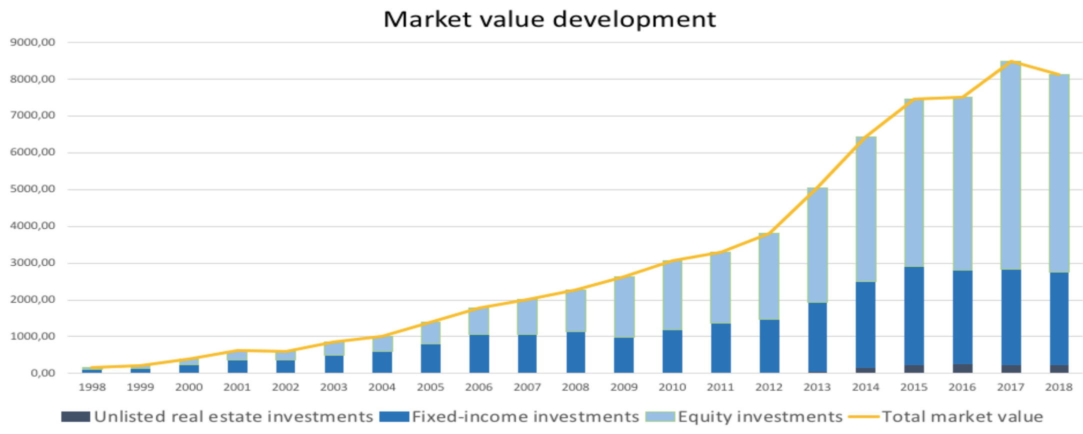

Figure 1: The market development in the Norwegian Pension Fund from 1998 to 2018.

The GPFG is managed by Norges Bank Investment Management and Folketrygdfondet, respectively, under mandates laid down by the Ministry of Finance (Norwegian Ministry of Finance, 2017). All investments in the Fund are made global and are invested outside Norway to reduce the risk. The Fund is a well-diversified portfolio across different asset classes, countries and sectors. The portfolio is based on three different investment strategies; fund allocation, asset strategies and company investment. The Fund is invested in Asia, North America, Europe, Oceania, Latin America, Africa and the Middle East. The investments are measured against a benchmark index, which is set by the Ministry of Finance on the basis of indices from FTSE Group and Bloomberg Barclays Indices, where the strategic benchmark indices are divided into 30% fixed income and 70% equity investments.

The main research topic of this thesis is efficient asset allocation, where applying better inputs in the mean-variance framework will reduce the realized variance by increasing the reliability of the diversification effects, and as a result one will know which risks to expect. We will limit our research to only look at the equity investments of the Fund and compare the results with the FTSE benchmark index. Our analysis will be entirely concentrated on the financial performance of the Fund, and any political or other non-financial values will not be considered or discussed in the thesis. The fundamental aspect in this approach is the predictions for return, variance and correlation coefficients by use of the historical price series. The model will be programmed in Python and assessed with performance test statistics like the Sharpe-ratio, since the goal of this thesis is to assess if it is possible to improve and select the optimal model. However, it goes beyond this thesis to fully explain the causes of risk, and to elaborate on the changes in valuations of specific risks which aggregate in portfolio theory.

The remaining part of this thesis is organized as follows: The second chapter is a literature review and the in third we elaborate the most relevant theories for our thesis. The fourth chapter outlines the methodology in detail and discusses the estimation procedures used. Chapter five explains our data sources, the data collection and also the software used to apply the research methodology. While the sixth chapter shows the analysis, where we discuss our results and findings. The last and seventh chapter concludes our findings.

2. Literature review

There exist numerous studies researching asset allocation of funds. However, we have chosen to only discuss the literature we find relevant for our research of asset allocation of the Norwegian Government Pension Fund – Global. The central part of this literature review will be on existing literature on modern portfolio theory, portfolio construction, geographical diversification, market efficiency, systematic risk factors among many.

2.1 Modern portfolio theory

In 1952, Harry Markowitz published a paper on Modern portfolio theory, where he proved that the saying “Don’t put all your eggs in one basket” is true. Before the paper was issued, people had an intuitive sense that they should not put too much of their total wealth in a single investment or type of asset. Markowitz was therefore the first person to prove mathematically, that it was a question of how many eggs to put into which basket. Modern portfolio theory attempts to find a combination of assets which maximizes the expected return of a portfolio for a given level of risk, or similarly minimizes the variance of a portfolio for a given amount of expected return (Markowitz, 1952). The rationale behind this theory is that investors are risk-averse and will therefore only choose a riskier portfolio, if they will be compensated by a higher expected return. The following mathematical rules supports this theory:

1. The expected return on the portfolio is a weighted average of the expected returns on individual securities
2. The variance of return on the portfolio is a function of the variances of, and the covariances between, securities and their weights in the portfolio.

The mathematical calculations in modern portfolio theory are a way to structure and discipline your thinking as a portfolio manager – in a way to reduce risk and improve overall return (Hudson-Wilson, 1990). Hudson-Wilson also states that the more advanced our thinking can become, the higher return we will be able to achieve. In his article, Markowitz (1952) showed that assets in a portfolio can be combined to provide an "efficient" portfolio. By doing this, on can achieve the highest possible level of portfolio return for any level of portfolio risk, measured by the variance or standard deviation. These portfolios are then combined to generate the "efficient frontier". According to the investor's preferences, portfolios which have a combination below this efficient frontier will not maximize the efficient trade-off. Having established an efficient frontier, it is now necessary to decide where along the frontier the investor will choose a portfolio.

2.2. Portfolio construction

The main key in portfolio construction is how many properties an investor should hold to diversify risk. Risk is divided into non-systematic and systematic risk which can and cannot be diversified, respectively. Statman (1987) undertook a study to show how many stocks are needed in order to diversify all the nonsystematic risk and only be left with the market risk. Figure 2 show the average standard deviations of equally weighted portfolios by random selection as a function of the number of stocks. Statman proved that on average, portfolio risk does fall with diversification, but the power of diversification to reduce risk is limited by common risk or market risk.

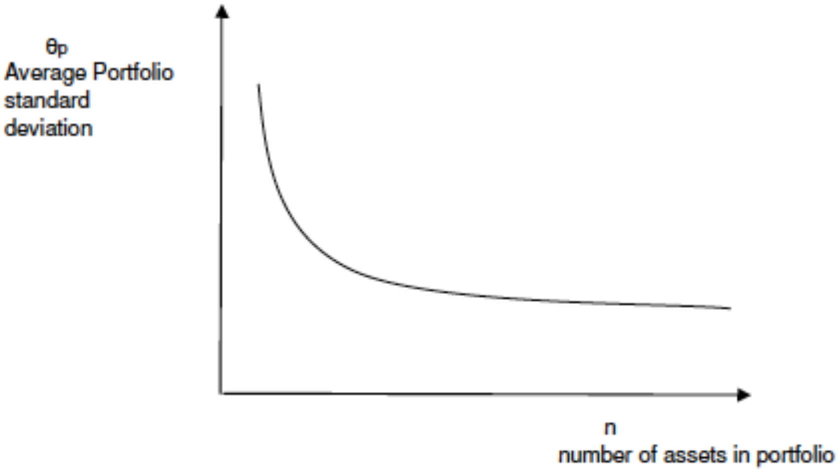

Figure 2: Portfolio risk decreases as diversification increases

Once the portfolio has been constructed the investor must measure and evaluate its actual behavior in relation to the expected performance. Such action usually involves selling certain investments and using the proceeds to acquire other investments for the portfolio. Portfolio Management therefore involves not only selecting a compatible group of investments that meet the investor's objectives, but also monitoring and restructuring the portfolio as dictated by the actual behavior of the investments.

2.3 Geographical diversification

A highly discussed topic in portfolio management, are the challenges and benefits from using diversification as one of your investment strategies. This is an extremely relevant subject for the GPFG. When deciding where to invest there are several different factors that can cause risk or lead to variability in returns on your investment, and there exist many circumstances that may influence your investment. Factors such as uncertainty of income, interest rates, inflation, exchange rates, tax rates, the state of the economy, default risk and liquidity risk. One way to control portfolio risk is using diversification. Diversification is when investments are made in a wide variety of assets so that the exposure to the risk is reduced (Brentani, 2004).

One way of diversifying is to use geographical diversification. That is the practice of diversifying an investment portfolio across countries, or over different geographical regions. Based on the premise that markets in different parts of the world are not highly correlated, one uses diversification to reduce the overall risk, and to improve the returns on the portfolio. In essence one wants to safeguard the portfolio investments from political turbulence and potential recessions among many aspects. However, diversifying your investments cannot eliminate or reduce all risk in your portfolio, because mostly all securities are affected by common (risky) macroeconomics factors. One cannot eliminate all exposure to general economic risk, however it is possible to reduce the exposure to certain factors by using geographical diversification (Bodie, Kane, & Marcus, 2012). In their research, Morck & Yeung (1991), Bodnar, et al. (1999), and Allayannis and Weston (2001) all found positive value effects from geographical diversification.

2.4 Market efficiency

A highly debated topic is the efficient market hypothesis (EMH), and the academic research on this area is extensive. Fama (1991) defined the efficient market hypothesis to be that security prices fully reflect all available information. However, Jensen (1978) has a refined version of the hypothesis which states that prices reflect information to the point where the marginal benefit, and hence the profits do not exceed the marginal costs. This definition implies that investors cannot achieve a return over the average without assuming above-average risk (Malkiel, 2003).

Furthermore, when it comes to using an active strategy that generates excess return, it often entails an investment strategy based on exploiting inefficiencies and mispricing in the market. This implies that these three elements are essential; interpretation of the EMH, the existence and identification of possible inefficiencies. Lakonishok et al. (1994) argues that anomalies are evidence of inefficiency and a potential to generate excess return with active management. On the contrary, Fama and French (1993) argues that anomalies and such inefficiencies are sources of risk premium and claim that these patterns of return may be consistent with an efficient market in which expected returns are consistent with risk. This is in accordance with more recent literature.

2.5 Systematic risk factors

The efficient market hypothesis is an underlying assumption for the well-known capital asset pricing model (CAPM). The main assumption in the CAPM is that the systematic risk of a security depends on the co-variation between the return on the security and the return on the market portfolio, measured by β (Sharpe 1964, Lintner 1965, Black 1972). That said, more recent empirical research has shown that the relationship between risk and return is more complex than assumed by the CAPM. Arbitrage pricing theory (APT) was introduced by Stephen A. Ross in 1976 and is a testable alternative to the CAPM. This is a theory that provides a solid theoretical framework for ascertaining whether multiple factors are "priced", i.e. are associated with a risk premium. Chen, Ross and Roll (1986) used data for individual equities during the period from 1962 to 1972 and concluded that at least three factors are definitely present in the prices.

There is extensive academic literature about which factors are associated with a persistent risk premium. Fama and French (1992) introduced two systematic risk factors in addition to the market factors in their so-called "three-factor-model". Their research was based on U.S. stocks during the period from 1963-1990, and they found out that a size factor (small versus large capitalization) and a value factor (value versus growth stocks), are additional determinants of stock returns. A further expansion of the model was made by Mark. M Carhart (1997) by adding a fourth factor capturing the one-year momentum anomaly. Another researcher, Cochrane (2011) argues that there exist dozens of priced factors that describe the cross-sectional variation in expected returns. He further argues that characterizing risk premium variation has replaced efficiency as the central organizing question of asset pricing research.

2.6 Active management and excess return

In financial literature, there are several studies that investigate the benefit of active management. One is market efficiency, this describes investors who "chase" alphas by uncovering inefficiency priced asset in order to achieve excess return. The theories provide the framework for organizing asset-pricing research. However, more recent literature explains many of these inefficiencies as priced systematic risk factors and how investors need to understand these factors in order to outperform the benchmark. There are two fundamental approaches to implement active management and, in that way, deviate from the benchmark portfolio, they are market timing and stock selection. These assume that priced systematic risk factors are determinants of stock returns.

2.6.1 Market timing

Market timing is the decision to change the proportion of the benchmark itself. This can be done in two ways; the first alternative is to shift some of the investment from the benchmark into a riskless asset, and the second alternative is to borrow and buy more of the benchmark.

In relevant literature on active management and performance, several researchers do not support the hypothesis that mutual fund managers are able to beat the market. Henriksson and Merton (1981) was one of them, he stated that managers are not able to follow an investment strategy that successfully times the return on the market portfolio. Their research emphasizes that the ability to earn superior returns are based on superior forecasting ability, and it would be a violation of the efficient market hypothesis with significant implications for the theory of finance.

Becker, Ferson Myers and Schill (1999) has done a more recent study based on more than 400 mutual funds in the time period from 1976 to 1994, where they distinguish timing based on publicly available information from timing based on finer information. They discovered that the average timing performance of mutual funds is insignificant and sometimes even negative.

2.6.2 Stock selection

Stock selection is when the manager chooses to hold securities in different proportions than the capital weights. By using a benchmark based on the characteristics held by 125 portfolios in mutual funds in the period from 1975 to 1994, Daniel et al. (1997) applied new measures of portfolio performance. Based on the benchmarks, “characteristic selectivity” and “characteristic timing” measures are developed to detect whether portfolio managers successfully time their portfolio weightings on these characteristics. Another part of the study examines whether portfolio managers can select stocks that outperform the average stock having the same characteristics. The research shows that mutual funds, particularly aggressive-growth funds, exhibit some selective ability, yet the funds exhibit no characteristic timing ability.

3. Theory

3.1 Modern portfolio theory

Markowitz made a difference between efficient and inefficient portfolios and proposed an optimization framework by geometrical analysis. In more mathematical terms and matrix notation, we can find the minimum portfolio variance, σ_P^2, for any particular portfolio return, μ_P. The weights, W_i, invested in each asset, assuming N different assets exist, is limited to 1.

$\sum_{i=1}^{N} w_i = 1$ (3.1)

The weights are a (N x 1) vector, $\boldsymbol{w} = \begin{pmatrix} w_1 \\ w_2 \\ \cdots \\ w_N \end{pmatrix}^2$

The portfolio return, r_p, is the weighed sum of the individual asset returns, $\boldsymbol{r}$, where r_p is a (1 x 1) scalar and $\boldsymbol{r} = \begin{pmatrix} r_1 \\ r_2 \\ \cdots \\ r_N \end{pmatrix}$ is a (N x 1) vector of returns.

$$r_p = w'r = (w_1, w_2, \ldots, w_N)\begin{pmatrix} r_1 \\ r_2 \\ \cdots \\ r_N \end{pmatrix}$$

$$= w_1 r_1 + w_2 r_2 + \ldots + w_N r_N \qquad (3.2)$$

This gives us the expected portfolio return,

$\mu_P = E[r_p] = w'E[r_p] = w'\mu$, where $\mu = E[r]$ (3.3)

The portfolio variance is given by,

$\sigma_P^2 = \boldsymbol{w'\Sigma w}$ (3.4)

$$= (w_1, w_2, \ldots, w_N)\begin{pmatrix} Var(r_1) & Cov(r_1, r_2) & \cdots & Cov(r_1, r_N) \\ Cov(r_2, r_1) & Var(r_2) & \cdots & Cov(r_2, r_N) \\ \vdots & \vdots & \ddots & \vdots \\ Cov(r_N, r_1) & Cov(r_N, r_2) & \cdots & Var(r_N) \end{pmatrix}\begin{pmatrix} w_1 \\ w_2 \\ \cdots \\ w_N \end{pmatrix}$$

Where $\boldsymbol{\Sigma}$ is a (N x N) covariance matrix containing the variance of all N assets returns and their pair wise covariance between the N assets returns. The minimum variance for a target portfolio return, μ^*, can be found by solving this quadratic function,

$$\text{Min } 0.5\sigma_P^2, \text{ with respect to } \mu_P = \mu^* \text{ and } \sum_{i=1}^{N} w_i = 1 \qquad (3.5)$$

By solving this problem, you get the optimal asset allocation weights that will minimize the risk for a given level of return. This optimal solution lies on the efficient frontier described by Markowitz (1952).

3.1.1 The efficient frontier

The efficient frontier is a graph representing a set of portfolios that maximize expected return at each level of portfolio risk (Bode Z., 2003). Plotting the efficient frontier is very complex. It is necessary to calculate the future expected returns and standard deviation, along with the correlation coefficients between each pair of assets.

The efficient frontier describes the collection of portfolios (i.e. asset mixes) that produces the highest expected return at various levels of risk as measured by the standard deviation of portfolio returns (Hudson-Wilson, 1990). Such portfolios can be seen as efficiently diversified. Figure 2 below, illustrates the graphical relationship of the individual assets and the efficient frontier. The expected return and standard deviation combinations for any individual asset end up inside the efficient frontier, because single asset portfolios are inefficient, in other words, they are not efficiently diversified. However, as you can see in figure 3 an investor can immediately discard portfolios below the minimum variance portfolio. The minimum variance portfolio is dominated or favored by portfolios on the upper half of the frontier because they yield a higher expected return with equal risk. Therefore, investors should only consider portfolios on the efficient frontier above the minimum – variance portfolio.

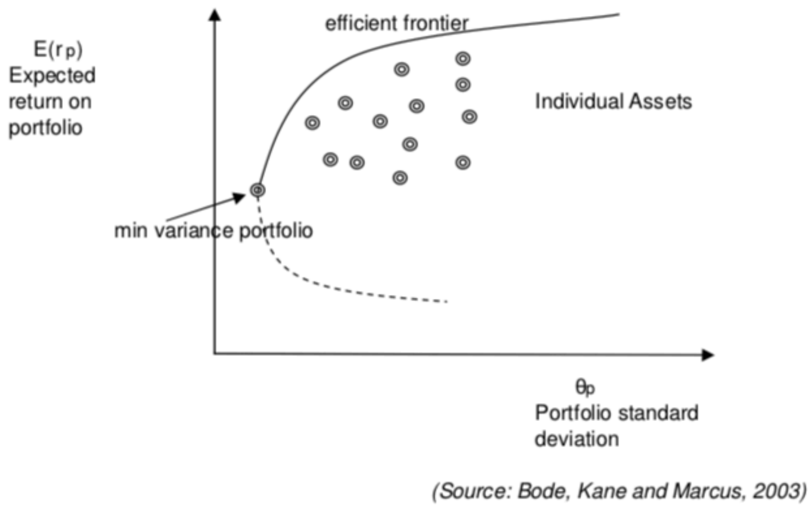

Figure 3: The graphical relationship of the individual assets and the efficient frontier.

When we optimize every single portfolio by means of the modern portfolio theory and plot the results in a risk-return space, we will obtain a combination of optimal portfolios which will form a hyperbola. The upper part of this hyperbola is dubbed the efficient frontier. When a risk-free asset is included, the efficient frontier will no longer be a set of portfolios, but one specific portfolio of risky assets, called the tangency portfolio (sometimes also called the optimal risky portfolio). That is the portfolio that tangents the efficient frontier when you draw a line from the risk-free asset to the efficient frontier in a (μ_P, σ_P)-space. The tangency portfolio together with the risk-free asset, will be the best fit for each investor's individual risk tolerance. And the portfolio return, r_P is,

$$r_P = \text{w}\, r_t^* + (1\text{-w})\, r_f \qquad (3.6)$$

Where r_t^* denote return from the tangency portfolio, w is the weight invested in the tangency portfolio and r_f denotes the return on the risk-free asset. r_P is called the best possible capital allocation line (CAL). Due to the fact that the variance and the risk of a risk-free asset is zero, the variance of this portfolio will be,

$$\begin{aligned} \sigma_P^2 &= w^2\sigma_P^2 + (1-w)^2\,\sigma_{rf}^2 + 2w^2\,(1-w)^2\,cov(r_t, r_f) \\ &= w^2\sigma_P^2 + (1-w)^2\,0 + 2w^2\,(1-w)^2\,0 \qquad (3.7) \\ &= w^2\sigma_P^2 \end{aligned}$$

Now that we know the portfolio return and we know that the standard deviation is the square root of the variance, we can compute the CAL as,

$$r_P = r_f + \left(\frac{r_t - r_f}{\sigma_t}\right)\sigma_p \qquad (3.8)$$

3.2 Capital Asset Pricing Model (CAPM)

The Capital asset pricing model, referred to as CAPM, was developed from modern portfolio theory and is one of the most popular tools for quantifying and measuring risk for equities. The model relates the required rate of return on a security to its systematic risk as measured by beta, and the beta is estimated using a regression of the portfolio returns in excess of the risk-free rate on the benchmark returns (Risk and Return, NBIM 2016). The CAPM predicts the relationship between the risk and equilibrium expected returns on risky assets (Bodie, Kane, Marcus, 2013). Systematic risk is non-diversifiable risk; therefore, beta is effectively measuring the systematic risk of a specific asset. The CAPM's expected return/beta relationship is as follows:

$$E[r_A] = r_f + \beta_A[E(r_m) - r_f] \qquad (3.9)$$

Where, $E[r_A]$ = Expected return of Asset A

r_f = Risk-free rate of return

β_A = Contribution of Asset A to the risk of a portfolio

$E(r_m)$ = Expected return of the market

Using the model requires certain assumptions and simplifications about the market and the investors. Assumptions such as, investors are risk averse and maximize expected utility, or that investors choose portfolios on the basis of their expected mean and variance returns among many. One of the forecasts of the CAPM is that in equilibrium, all assets should lie on the security market line. If the investment is located above the security market line, the investor will choose to invest because the return is higher than what is required for its level of risk. Or if the investment is located below the security market line, the investor will choose not to invest because the return is too low (Brentani, 2004). The security market line is defined as a visualization of the CAPM.

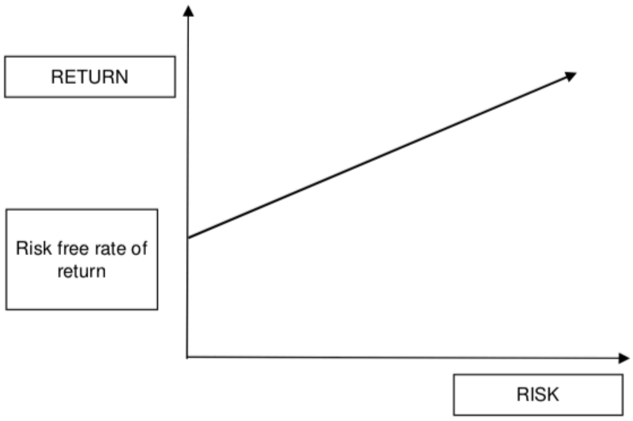

Figure 4: Capital asset pricing model with the security market line

3.3 Risk aversion and utility functions

We need to define a measure of preference towards risk, which allows us to rank portfolio decisions in order to model the strategic asset allocation problem. According to Jehle (2000) these preferences can be represented by a utility function, U(W), this is an analytical device summarizing the information contained in the investor's preference relation. The concept is used as preferences over state-dependent return profiles. The basis of the concept is that higher preferred wealth levels will result in a higher utility value, which will naturally make the function strictly increasing in terminal wealth. If a zero-mean risk opportunity exists, often called a fair game, and the investors prefers her terminal wealth rather than entering into a game with risk, she is called a risk-averse investor. If the investor is risk-loving, she prefers the game, and if she is indifferent, she is said to be risk-neutral.

In terminal wealth U(W) is increasing and is decreasing when U'(W) > 0 and U''(W) < 0. This yields a concave function, where the extent of the concavity measures the risk aversion of the investor. Arrow-Pratt risk measure (Pratt, 1964; Arrow, 1971) define the absolute risk aversion (ARA) as the negative of the second derivative of the utility function, scaled by the first derivative, and are defined as

$$ARA(W) = \frac{U''(W)}{U(W)} \qquad (3.10)$$

The coefficient quantifies the aversion to a zero-mean risk around W, indicating the aversion towards an absolute sized risk.

Relative risk aversion (RRA) is another risk measure, and indicates the investor's willingness to pay to avoid a gamble of a certain size relative to W. RRA can therefore be defined by taking investors wealth into account

$$RRA(W) = \frac{WU''(W)}{U'(W)} \qquad (3.11)$$

We desire to see decreasing absolute risk aversion in wealth and constant relative risk aversion.

3.4 Short-term portfolio choice and asset allocation

In portfolio management the common practice is a top-down approach when it comes to asset allocation. The first step is to decide on the weights of the country allocation. Further, step two involves the choice of stocks and their weights in the countries under consideration. This is a well-known method to diversify portfolios, since financial markets in different parts of the world are often not highly correlated with one another. For example, if the developed markets are declining because of recession in the economy, it can be more valuable to allocate part of this portfolio to emerging economies with higher growth rates such as China, Brazil and India. Green and Hollifield (1992) argued that if stocks or indices are highly correlated, and exhibit a high diversity of betas, then we can form portfolios with essentially zero factor risk. However, such a portfolio will take a large negative position in one stock and an even larger positive position in another stock, or indices.

Markowitz (1952) mean-variance analysis is built on the theory that investors should choose assets if they care only about the mean-variance, or equivalently the mean and standard deviation - of portfolio returns over a single period. For simplicity, he used three assets: stocks, bonds and a short-term money market fund. As you can see in the figure below, the vertical axis shows expected return, and the horizontal axis shows risk as a measured by standard deviation.

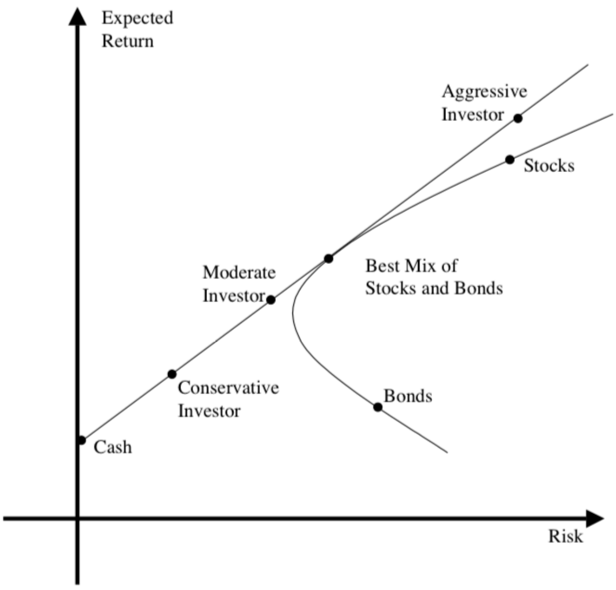

Figure 5: Mean-standard deviation diagram (Markowitz, 1952).

As you can see, stocks have high standard deviation, and therefore high expected return, while bonds are low. The curved line shows the set of means and standard deviations that can be achieved by combining stocks and bonds in a risky portfolio. A risk averse investor would choose a point on the straight line, which is the mean-variance efficient frontier.

To make a short-term portfolio choice, NBIM must choose the weights on the risky assets. In a simple case with two assets, where one asset is riskless with simple return Rf_{t+1} from time t to time t + 1, and the other asset is risky with R_{t+1} from time t to time t+1, with conditional mean $E_t R_{t+1}$ and conditional variance σ_t^2. The risk-free interest rate is realized at t + 1, and known one period in advance at time t. The conditional mean and variance are the mean and variance conditional on the investor's information at time t; thus, they are written with t. The investor puts a share α_t of her portfolio into the risky asset. Then the portfolio return is

$$R_{p,t+1} = \alpha_t R_{t+1} + (1 - \alpha_t) R_{f,t+1}$$

$$= R_{f,t+1} + \alpha_t (R_{t+1} - R_{f,t+1}) \qquad (3.12)$$

With the mean portfolio return

$$E_t R_{p,t+1} = R_{f,t+1} + \alpha_t (R_{t+1} - R_{f,t+1}) \qquad (3.13)$$

and the variance is

$$\sigma_{pt}^2 = \alpha_t^2 \sigma_t^2 \qquad (3.14)$$

The preferred investment for the NBIM is a high mean and a low variance of portfolio returns. We assume that these trade-offs are linear, which means that she maximizes a linear combination of mean and variance, with a positive weight on mean and a negative weight on variance:

$$\max_{\alpha_t} \left(E_t R_{p,t+1} - \frac{k}{2}\sigma_{pt}^2 \right) \qquad (3.15)$$

Then, substituting in the mean and variance of portfolio returns, and subtracting $R_{f,t+1}$, which can be written

$$\max_{\alpha_t} \ \alpha_t(E_t R_{p,t+1} - R_{f,t+1}) - \frac{k}{2}\alpha_t^2 \sigma_{pt}^2) \qquad (3.16)$$

And the solution to this maximization problem is

$$\alpha_t = \frac{E_t R_{p,t+1} - R_{f,t+1}}{k\,\sigma_t^2} \qquad (3.17)$$

This formula tells us that the portfolio share in the risky asset should equal the expected excess return, also called risk premium, divided by conditional variance times the coefficient k that represents aversion to variance. However, for NBIM there will be many risky assets, and the definition of the portfolio return is the same, except the denotation of vectors and matrices. Thus, R_{t+1} is now a vector of risky returns with N elements. The mean vector is $E_t R_{\,t+1}$ and a variance-covariance matrix Σ_t . Also, α_t is now a vector of allocation to the risky assets. So, the maximization problem now becomes:

$$\max_{\alpha_t} \ \alpha'_t(E_t R_{\,t+1} - R_{f,t+1}\iota) - \frac{k}{2}\alpha_t' \Sigma_t \alpha_t) \qquad (3.18)$$

Here ι is a vector of ones, and $(E_t R_{\,t+1} - R_{f,t+1}\iota)$ is the vector of excess returns on the N risky assets over the riskless interest rate. The variance of the portfolio return is $\alpha_t' \Sigma_t \alpha_t$. The solution to this maximization problem is

$$\alpha_t = \frac{1}{k}\Sigma_t^{-1}(E_t R_{\,p,t+1} - R_{f,t+1}\iota) \qquad (3.19)$$

The single excess return is replaced by a vector of excess returns, and the reciprocal of variance is replaced by Σ_t^{-1}, the inverse of the variance-covariance matrix of returns. The scalar term $\frac{1}{k}$ is the investor's preference. Thus, investors differ only in the overall scale of their risky asset portfolio, not in the composition of the portfolio. Tobin (1958) and his mutual fund theorem says that conservative investors with a high k hold more of the riskless asset and less of all risky assets, but they do not change the relative proportions of their risky assets, which are determined by the vector $\Sigma_t^{-1}\,(E_t R_{p,t+1} - R_{f,t+1}\iota)$.

3.5 Volatility

Volatility is the main measurement of risk and measures the spread in returns for a given security or market index. According to the rational expectation model, market excess return and market volatility is positively correlated over the long-run in the cross-section of assets. The rational expectation model states that investors should receive a risk premium for taking on risk, i.e. the higher the volatility the higher excess return demanded. So, the higher the volatility, the higher the risk.

French et al. (1987) argued that the negative relationship between market excess return and market volatility exist because excess return is positively correlated to expected volatility. However, volatility is highly persistent, so an increase in unexpected volatility would increase the future expected risk premium, hence, decrease the current stock price.

3.6 Sharpe ratio

The Sharpe ratio is the slope of the capital allocation line (CAL), and was developed by William F. Sharpe (1966). There are two essential versions from Sharpe (1994), ex-ante Sharpe ratio, which uses expected portfolio return in the calculations, and ex-post Sharpe ratio, which uses realized portfolio return. The Sharpe ratio aim to measure risk-adjusted performance by subtracting the risk-free interest rate from the portfolio rate of return, such that we get excess return of the portfolio, and then divide excess return by the standard deviation of the portfolio returns.

$$\text{Sharpe Ratio} = \frac{rp - rf}{\sigma p} \qquad (3.20)$$

rp = The observed average return

rf = The average risk-free return

σp = The standard deviation of fund returns

The Sharpe ratio is popular in finance due to its simplicity and its ability to measure the tradeoff between risk and return. It follows the ideology of the rational expectation model in that an investor should be properly compensated for taking on additional risk. If the excess return on the investment is relatively low with respect to the risk, then the Sharpe ratio would be low. We want as high value as possible in the Sharpe ratio, just as we want as high α-value as possible in the Single Index Model.

A drawback with the Sharpe ratio is that it includes standard deviation of excess return, which assumes that the excess return in the portfolio follows a normal distribution. Therefore, kurtosis and skewness can decrease the accuracy of the Sharpe ratio. The standard deviation is measured by the distance each return has from the mean, so a large observed return, positive or negative, in a series of relatively small returns will penalize the Sharpe ratio. An example from Harding (2002) is that a suddenly large positive return in a series of small, consistent and positive returns will generate a lower Sharpe ratio, due to the increased standard deviation. One solution to this problem is to use the Sortino rate, which produce a semi-standard deviation based on only negative returns to use in the denominator instead of standard deviation. Another flaw in an ex-ante Sharpe ratio is the estimation, if the estimates are spurious then the Sharpe ratio will be spurious.

4. Methodology and Model

In this chapter we will explain the methodology we have applied to make our estimations and assumptions about asset allocation of the Norwegian Government Pension Fund - Global. Additionally, we will elaborate on what models we have used in order to interpret these findings and estimations.

To outline the model specifications and assumptions the following structure is applied. At first the models we used are considered briefly, and secondly, we will test if there are any differences between the models. Thirdly we make a comparison between the benchmark, the minimum variance portfolio and the Sharpe ratio portfolio. Then, at last we will elaborate on the distributions used in determining the optimal allocation. The Python source code used to compute our empirical results is given in the appendix. This thesis excludes transaction cost and tax from the portfolios, because transaction cost is negligible, especially for institutional investors. Tax is constantly changing and is different from country to country, since tax often depends on the level of capital, and because dividend- and capital yield can have different tax rates too.

4.1 Models implemented and estimation process

The thesis is built on the perspective of a utility maximizing investor, with a main focus on maximizing the portfolio risk-adjusted return of the portfolio. Here, the return is measured and ranked by the Sharpe ratio and the minimizing variance. The dynamic asset allocation strategy on the portfolio can be considered as the "optimized" portfolio, in the sense that they aim to have the same characteristics as Markowitz Minimum Variance Portfolio. That is, their goal is to earn a high Sharpe ratio compared to the benchmark, by minimizing the portfolio risk. They do so by rebalancing the portfolio every month. In response they reduce the weights in the equally weighted portfolio, when the risk increases and respond by increasing the weights in the equally weighted portfolio when the risk declines. The two portfolios made in this thesis are made on the constraint of "long only", meaning borrowing and short sale is restricted.

It is assumed that risky-asset returns at time t, and follows a random walk which are given by:

$$r_t = \mu + \varepsilon_t \qquad (5.1)$$

Where r_t is a (N x 1) vector of returns at time t, μ is a (N x 1) vector of mean returns, and ε_t is a (N x 1) vector of random shock at time t, that is an i.i.d random variable with zero mean and constant variance. The rolling window covariance matrices are computed using the following method, demonstrated with two assets, where historical covariance between daily returns from asset i and daily returns from asset j is applied:

$$= \frac{\sum_{t=1}^{T} \mathrm{r}_{i,t}\mathrm{r}_{j,t}}{T} \qquad (5.2)$$

Where T is the length of the rolling window. The constant mean return estimate, μ, is given as:

$$\hat{\mu}_t = \frac{1}{T} \sum_{i=t-T}^{t-1} \mathrm{r}_i \qquad (5.3)$$

Where T will be the length of the sample period, which is the distance from 01.08.2008 to 01.04.2018. The computation of the portfolios is done in matrix - and vector-form and is equivalent to the notation in equation (3.1) to (3.4).

4.1.1 Constant Expected Return (CER) analysis

The assumption in the Constant Expected Return (CER) model is that an asset's return over time is normally distributed with a constant mean and constant variance. The CER model constitutes the simplest specification of our general statistical model for asset returns. The model allows the returns on different assets to be simultaneously correlated, although the correlations are constant over time. Returns are independent over time both across assets and within the same asset.

The CER model in which $E[r_{t+1}|\mathcal{L}_t] = E[r_{t+1}] = \mu$, implies that each equation contains only one common regressor: a vector of ones. In this case we have for the i-th return:

$$y_i = e_T\delta_i + u_i,$$

Where, $$y_i = \begin{bmatrix} y_{i,1} \\ y_{i,2} \\ \vdots \\ y_{i,T} \end{bmatrix}, \qquad X_i = e_T = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}. \qquad (5.4)$$

The OLS/SURE estimates of the relevant parameters are then simply

$$\hat{\delta}_\iota = \frac{1}{T} \sum_{t=1}^{T} r_{i,t} = \overline{r_\iota} \qquad (5.5) \qquad\qquad \hat{\sigma}_{11} = \hat{\sigma}_1^2 = \frac{1}{T} \sum_{t=1}^{T} (r_{i,t} = \overline{r_\iota})^2 , \quad (5.6)$$

which is sample mean and sample variance. Before we obtained any results, we had to simulate 50.000 different portfolios that would give us the most optimized asset allocation. To display this, we have in figure 6 below plotted the simulation with constant expected return for only 50 portfolios. This is to give a clearer overview of the distribution of the portfolios.

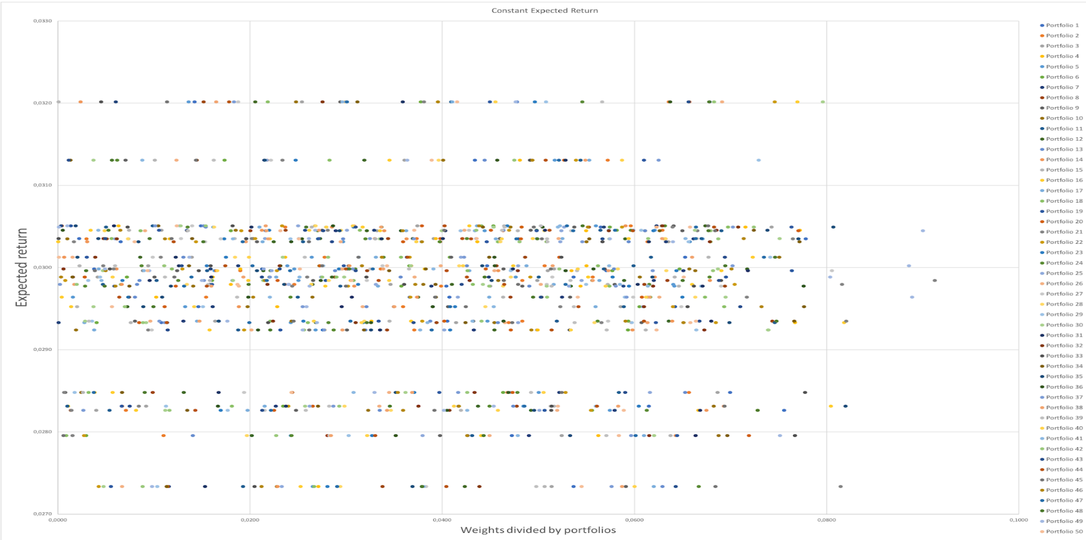

Figure 6: The distribution of portfolios with constant expected return

4.1.2 Time-Varying Expected Return (TVER) Analysis

When reviewing a time-varying expected return, we select time-varying predictors that requires using the properties of observed data to predict future observations of the relevant variables. We used the sum-of-part (SOP) approach, where the idea is to decompose returns in several parts, and to implement simple times series analysis. This is done to predict the individual component, and then generate a time-varying expected (predicted) return by aggregating predictions.

In chapter 4.1.1, we derived the tangency portfolio using the CER model and used the unconditional moments as inputs of the asset allocation optimization. While, when we used the TVER alternative we looked at univariate time series methods. We adopted the following specification for all countries stock market returns:

$$r_{t,t+36}^{Country} = \left(p_{t+36}^{Country} - p_t^{Country}\right) + \sum_{i=j}^{36} \frac{D_{t+j}^{Country}}{p_{t+j}^{Country}} \quad (5.7)$$

$$\left(p_{t,t+36}^{Country} - p_t^{Country}\right) = E_t\left(p_{t+36}^{Country} - p_t^{Country}\right) + u_{1,t+36}^{Country}$$

$$E_t\left(p_{t+36}^{Country} - p_t^{Country}\right) = \beta_0^{Country} + \beta_1^{Country}\left(p_t^{Country} - \frac{1}{36}\sum_{j=1}^{36} p_{t-j}^{Country}\right)$$

$$\sum_{j=1}^{36} \frac{D_{t+j}^{Country}}{p_{t+j}^{Country}} = E_t\left(\sum_{i=0}^{36} \frac{D_{t+j}^{Country}}{p_{t+j}^{Country}}\right) + u_{2,t+36}^{Country}$$

$$E_t\left(\sum_{j=1}^{36} \frac{D_{t+j}^{Country}}{p_{t+j}^{Country}}\right) = \left(\sum_{i=0}^{35} \frac{D_{t-j}^{Country}}{p_{t-j}^{Country}}\right)$$

In words, log-prices are mean reverting towards a trend that is estimated with the 3-year moving average of past prices. Here we also simulate 50.000 different portfolios to give us the most optimized asset allocation with the TVER method. In figure 7 below we plotted the simulation with time-varying expected return for only 50 portfolios, to give a clearer overview of the distribution of the portfolios.

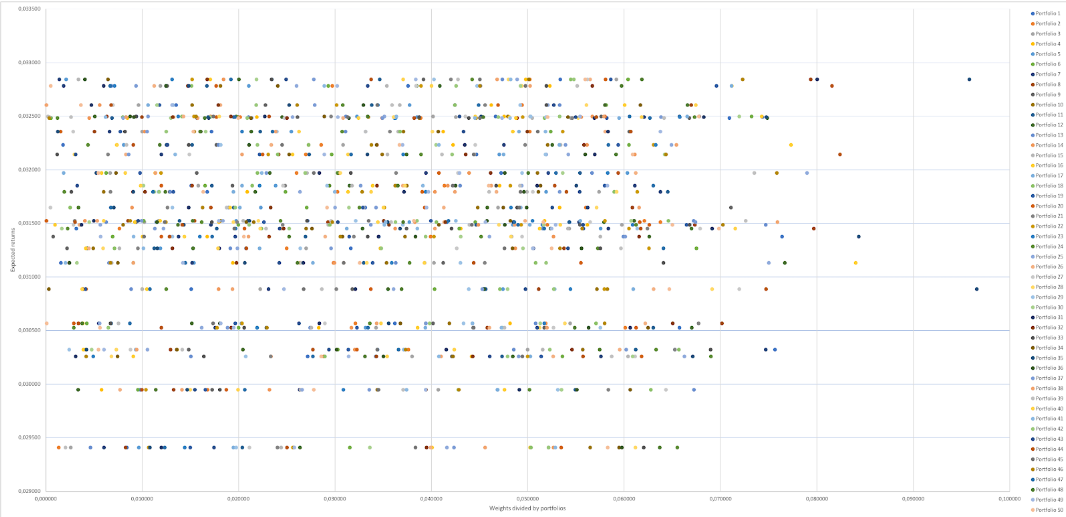

Figure 7: The distribution of portfolios with time-varying expected return

There are mathematical optimization techniques that would have resulted in the same result. However, we chose to use a Monte Carlo Simulation (a more manual method) to explain the whole concept of Efficient Frontier and optimal portfolios.

4.2 Portfolio performance measurement

4.2.1 Benchmark Comparison

A benchmark portfolio is an index created to include different securities representing some aspects of the total market. It is used to compare the allocation, risk and return of a given portfolio. The most obvious, conventional method entails a comparison of the managed portfolio against a broad market index. If an investment portfolio has gained greater returns than a benchmark portfolio during the same time period, then the portfolio is said to have outperformed the benchmark. We have used the GPFG's benchmark which is the FTSE.

Even though comparing a portfolio to a benchmark is common practice today, it is not without complications. After all, the risk of the investment portfolio and the benchmark index may not be the same. Therefore, the gain could come with higher risk. It means that if the portfolio has performed better than the benchmark portfolio, it may be a result of a managed portfolio being riskier than the benchmark portfolio. Consequently, that kind of comparison could lead to incomplete conclusions.

4.2.2 Mean return and standard deviation

Because the mean return is so applicable, it is useful to get an overall view of the empirical portfolios. The specific portfolio mean return is computed from equation (3.6), however, the portfolio mean return is annualized in this thesis by using this equation for simple interest:

$$\bar{\mu}_p = \hat{\mu}_p 12 \qquad (5.8)$$

Here $\bar{\mu}_p$ is the annual portfolio mean return, and $\hat{\mu}_p$ is realized monthly portfolio mean return. The variance is the spread of the observations and is computed from equation (3.7). The standard deviation is the square root of variance. To annualize the portfolios monthly standard deviation, the monthly standard deviation of a given portfolio, $\hat{\sigma}_p$, is multiplied with the square root of 12.

$$\bar{\sigma}_p = \hat{\sigma}_p \sqrt{12} \qquad (5.9)$$

Here $\bar{\sigma}_p$ is the annualized standard deviation of a given portfolio return. All result in this thesis are reported in annual terms, to simplify and avoid confusion.

4.2.3 Skewness

Skewness measures the deviation of symmetry in a dataset, if the dataset deviates to the left or to the right of the center point. A perfectly symmetric dataset, like the normal distribution, looks exactly the same on the right-hand side of the mean, as on the left-hand side of the mean. A dataset is symmetric if it has a skewness value of zero. The dataset has more values on the left-hand side of the mean if the skewness value is negative, meaning that the data are skewed to the left of the mean, and the left tail is longer than the right tail. Vice versa, if the skewness value is positive, then the right-hand side of the mean has a longer tail than the left-hand side of the mean, and the dataset is skewed to the right of the mean.
In our empirical portfolios, a negative skewness will indicate that the mass of the returns is concentrated to the right of the mean, the portfolio has a tail of returns

that are lower than the mean; investors do generally not prefer this. A positive skewness indicate that the mass of the returns is concentrated to the left of the mean. The portfolio has a tail of returns that are higher than the mean; investors generally prefer positive skewness above and beyond their preference for a higher mean and lower volatility. Note that portfolio skewness unequal zero implies that the portfolios are not normally distributed. Skewness has this formula:

$$S = \frac{\frac{\sum_{i=1}^{N}(r_i - \hat{\mu}_p)^3}{N}}{\hat{\sigma}_p^2} \qquad (5.10)$$

Where N is the number of returns in the portfolio. r_i, $\hat{\mu}_p$ and $\hat{\sigma}_p$ are monthly portfolio returns, monthly portfolio mean return and monthly portfolio standard deviation respectively, equivalent to previous notations.

4.2.4 Sharpe ratio

The Sharpe ratio is the main performance measurement in this thesis, since the ratio can compare portfolios with different exposure to risk. A rational investor will prefer the portfolio with the highest Sharpe ratio regardless of its limitations. In the evaluation process, we will use this version of the Sharpe ratio,

$$\widehat{SR}_p = \frac{\hat{\mu}_p - r_f}{\hat{\sigma}_p} \qquad (5.11)$$

5. Data sources and data collection

To illustrate what occurs in practice we consider a Norwegian investor from NBIM which sees the Norwegian 3-Month rate as the risk-free rate. The dataset in this thesis is based upon publicly available databases where we gathered data on 25 different countries consisting of each country’s 10-year benchmark bond, the Norwegian FIBOR 3-Month rate, consumer price index and the dividend yield for all countries. The risky assets available for portfolio allocation are Austria, Belgium, Canada, Czech Republic, Chile, China, Denmark, France, Germany, Hungary, Israel, Italy, Japan, Mexico, Netherland, Poland, Portugal, Russia, South Africa, South Korea, Spain, Sweden, Switzerland, United Kingdom and the United States.

Our data is at a monthly frequency and each variable consist of 118 observations. We have mainly collected data from sources such as Federal Reserve Economic Data, Bloomberg and Macrobond. All data is collected in US dollar. We will compare our asset allocation results with the Fund's benchmark index, set by the Ministry of Finance on the basis of indices from the FTSE. However, we did not have the opportunity to incorporate all the countries from the benchmark, because the availability of data for certain countries were limited. Therefore, we chose to eliminate these countries since we could not collect complete data for the specific time period. This is done to achieve a more thorough analysis. The data has been collected from July 31, 2008 until April 31, 2018. This coincides with the time period for the FTSE Benchmark Index, and the timespan that was available for the individual country's dividend yield, collected from Bloomberg.

5.1. Python

Python is a powerful programming language, that offers more flexibility and standard functions than the language and interface of Visual Basic available in Microsoft Excel. Based on this rationale, our supervisor strongly advised us to take the extra time and effort to make extensive use of Python in our research. Working on our thesis we have learned the language of programming in Python, and lengthy hours of debugging has enabled us to fully understand every facet of each model and its code.

In the Python source code, included in the appendix, you find the entire code we wrote to calculate the optima and minima and to estimate our models. For the estimation procedure, functions to conduct rolling windows and to loop estimation procedures, were programmed. The modules that are used to conduct the procedures of Modern Portfolio Theory make use of the formulas that are standard functionality in Python, just like the program used to conduct the test procedures. To further enhance our effort and obtain a cross reference, we used a framework written by Carlo A. Favero, a professor of Finance at Bocconi University, as inspiration and a guideline for our programming and estimation of the Dynamic Asset Allocation. However, his program is written in MATLAB and not Python.

6. Results and analysis

The results and analysis of the Fund with asset allocation, using both constant expected return and time-varying expected return, will be presented in this chapter. After extensive development of our models in Python, we can now use these models in practice and interpret the results. By applying different parameters to them and comparing these, we will be able to maximize the likelihood of finding the best combination of assets by configuring the model. As a result of trial and error, we will achieve the optimal Sharpe ratio and the minimum variance portfolio. After all, decisions for asset managers should be based on reliable and vast amounts of data and is dependent on the length of the investment horizon. As mentioned earlier, our analysis will only concentrate on financial performance and not political, economic or ethical factors. We will show the realized rolling Sharpe ratios over a measurement period of one and three years.

The results will be presented in four parts. We begin with presenting the basic features of our research with descriptive statistics to create an overview of the results. Following, a presentation of the result from the CER model from the time period 2008-2018, where we use an efficient frontier with expected returns and the expected volatility to visualize the optimal portfolio. Further, we present the same efficient frontier models and weights with the univariate time series for TVER model. Here we use a roll forward of three years that gives us the time period of 2011-2018. For both the CER and TVER models we present the portfolio weights calculated for the Sharpe ratio portfolio and then the minimum variance portfolio. At the end we will compare our results and weights with the weights from NBIM.

6.1 Descriptive statistics

In descriptive statistics we use figures and tables to describe the data and show general trends over time. In figure 8 below, we display the total return for each country with the CER method. This is provided to show an overview of the original dataset of all 25 countries from 2008-2018 and look for potential trends.

We can see that all the countries in this time period has a relative positive return, expect from a downturn in the period from 2008-2009 due to the financial crisis.

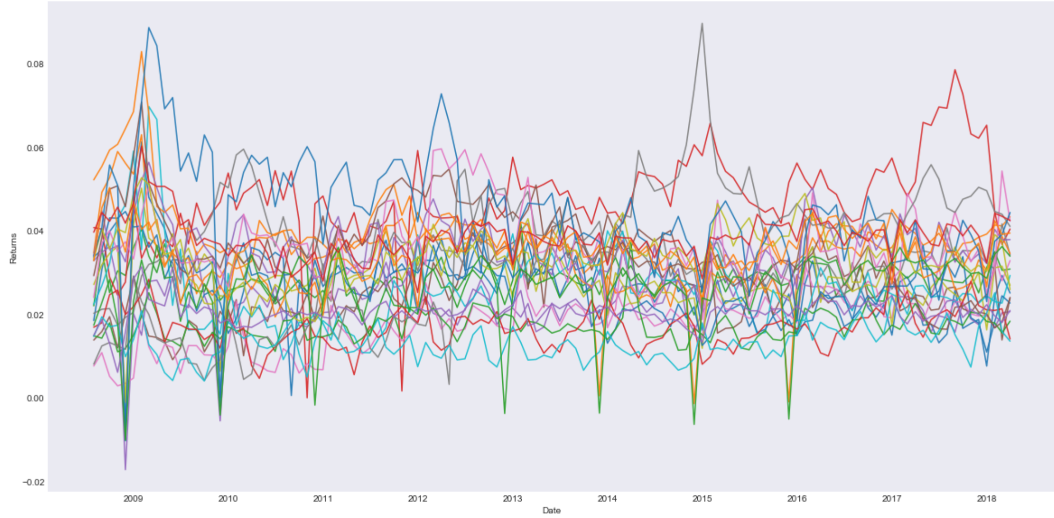

Figure 8: The total returns with CER

The second figure, figure 9 below, displays the total return for each country with the TVER method. We look for a general trend for the 25 countries of the second dataset where we roll forward 3 years, for the time period 2011-2018. Here we see the returns correlate around zero. Both figures confirm that there is no trend in the returns between 2008-2018. This is exactly the result one wishes for concerning returns, because returns are supposed to be fluctuating.

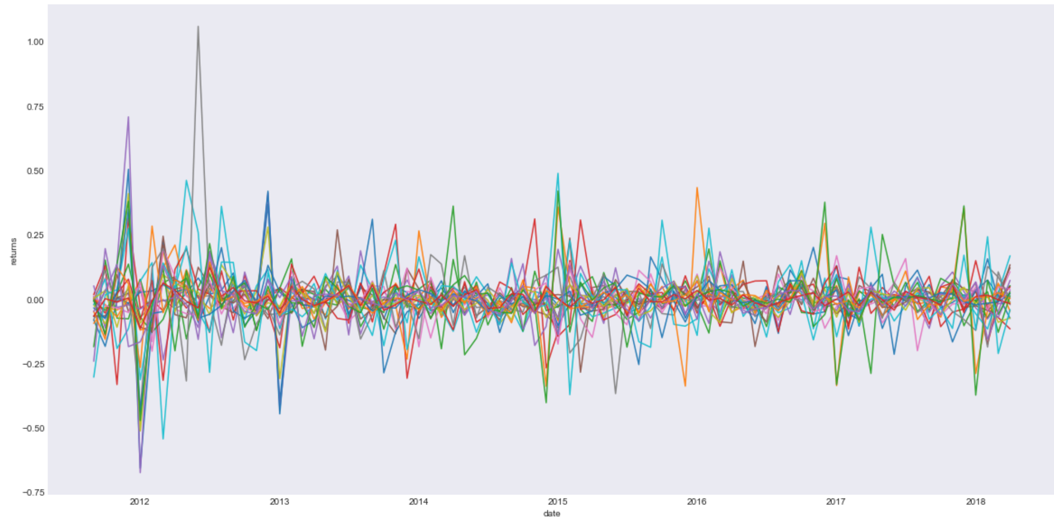

Figure 9: The total returns with TVER

The results in table 1 and 2 below, show the descriptive statistics of our simulation of the two portfolios between 2008-2018 and 2011-2018. In the table we see the variables for mean, standard deviation, min and max values, percentage quartiles for returns, volatility, Sharpe ratio and the weights of every country. The mean returns for the constant expected return portfolio and time-varying expected return portfolio are slightly different. The CER portfolio has a mean return of 2,98%, while the TVER portfolio has a mean return of 5,02%. The constant expected return portfolio has a standard deviation of 31,47% and the time-varying expected return has a standard deviation of 35,59%.

	Mean return	Standard Deviation	Minimum	25 %	50 %	75 %	Maximum	Skewness
Constant Expected Return								
Returns	0,029877	0,001004	0,025666	0,029199	0,029876	0,030553	0,034810	
Volatility	0,314795	0,022100	0,245128	0,299473	0,312874	0,328086	0,451823	
Sharpe Ratio	0,095302	0,006384	0,067726	0,090978	0,095412	0,099688	0,120403	
Austria Weight	0,039843	0,023115	0,000000	0,020131	0,039746	0,058782	0,126930	-0,084396
Belgium Weight	0,040076	0,023097	0,000001	0,020483	0,040137	0,058897	0,118681	1,372138
Canada Weight	0,040114	0,023166	0,000000	0,020441	0,040265	0,059155	0,116453	0,453655
Chech Republic Weight	0,039927	0,022980	0,000000	0,020450	0,039996	0,058555	0,119263	-0,903252
Chile Weight	0,039918	0,023030	0,000000	0,020407	0,039684	0,058688	0,115724	-2,701376
China Weight	0,040021	0,023015	0,000002	0,020599	0,040026	0,058748	0,118514	0,322688
Denmark Weight	0,039974	0,023198	0,000002	0,020141	0,039879	0,059012	0,124414	-0,123116
France Weight	0,040030	0,023055	0,000001	0,020599	0,039882	0,058856	0,114059	0,437883
Germany Weight	0,040012	0,023102	0,000000	0,020367	0,039920	0,058898	0,123255	-0,866413
Hungary Weight	0,039983	0,023059	0,000002	0,020363	0,040011	0,058801	0,116420	1,747667
Israel Weight	0,040136	0,023047	0,000001	0,020608	0,040220	0,058838	0,118350	-0,119147
Italy Weight	0,039931	0,023059	0,000005	0,020294	0,039956	0,058690	0,122602	0,413779
Japan Weight	0,039977	0,023088	0,000007	0,020530	0,039998	0,058917	0,121071	1,640928
Mexico Weight	0,039902	0,023003	0,000000	0,020397	0,039886	0,058694	0,118285	0,194599
Netherland Weight	0,040010	0,023073	0,000002	0,020473	0,039962	0,058920	0,118568	0,365510
Poland Weight	0,039987	0,023069	0,000003	0,020453	0,039891	0,058751	0,125376	0,468341
Portugal Weight	0,040143	0,023080	0,000003	0,020577	0,040219	0,059103	0,113733	0,710362
Russia Weight	0,039960	0,023021	0,000004	0,020400	0,039951	0,058649	0,124992	0,187509
South Africa Weight	0,039812	0,023094	0,000003	0,020214	0,039676	0,058754	0,129951	-0,005617
South Korea Weight	0,040121	0,023099	0,000009	0,020547	0,040169	0,059105	0,113348	0,293570
Spain Weight	0,040162	0,023057	0,000000	0,020694	0,040094	0,058949	0,112293	-0,214028
Sweden Weight	0,040024	0,023193	0,000000	0,020361	0,039984	0,058950	0,117331	-0,298157
Switzerland Weight	0,039866	0,023051	0,000001	0,020210	0,039735	0,058721	0,120694	-2,429658
UK Weight	0,040096	0,023118	0,000000	0,020350	0,040113	0,059014	0,136021	0,687719
US Weight	0,039977	0,023027	0,000000	0,020453	0,040043	0,058674	0,117503	-0,478349

Table 1: The total returns with CER

Time-Varying Expected Return	Mean return	Standard Deviation	Minimum	25 %	50 %	75 %	Maximum	Skewness
Returns	0,050237	0,012405	-0,000958	0,041952	0,050231	0,058567	0,106215	
Volatility	0,355995	0,034018	0,250767	0,332056	0,354305	0,377870	0,535542	
Sharpe Ratio	0,142118	0,036549	-0,002961	0,117479	0,141472	0,166162	0,302250	
Austria Weight	0,039843	0,023115	0,000000	0,020131	0,039746	0,058782	0,126930	-1,001920
Belgium Weight	0,040076	0,023097	0,000001	0,020483	0,040137	0,058897	0,118681	0,807365
Canada Weight	0,040114	0,023166	0,000000	0,020441	0,040265	0,059155	0,116453	-0,182389
Chech Republic Weight	0,039927	0,022980	0,000000	0,020450	0,039996	0,058555	0,119263	0,028626
Chile Weight	0,039918	0,023030	0,000000	0,020407	0,039684	0,058688	0,115724	0,151699
China Weight	0,040021	0,023015	0,000002	0,020599	0,040026	0,058748	0,118514	0,225776
Denmark Weight	0,039974	0,023198	0,000002	0,020141	0,039879	0,059012	0,124414	-0,421228
France Weight	0,040030	0,023055	0,000001	0,020599	0,039882	0,058856	0,114059	-0,423806
Germany Weight	0,040012	0,023102	0,000000	0,020367	0,039920	0,058898	0,123255	-0,932306
Hungary Weight	0,039983	0,023059	0,000002	0,020363	0,040011	0,058801	0,116420	0,008571
Israel Weight	0,040136	0,023047	0,000001	0,020608	0,040220	0,058838	0,118350	-0,075795
Italy Weight	0,039931	0,023059	0,000005	0,020294	0,039956	0,058690	0,122602	0,432155
Japan Weight	0,039977	0,023088	0,000007	0,020530	0,039998	0,058917	0,121071	0,524684
Mexico Weight	0,039902	0,023003	0,000000	0,020397	0,039886	0,058694	0,118285	-0,027485
Netherland Weight	0,040010	0,023073	0,000002	0,020473	0,039962	0,058920	0,118568	-0,054326
Poland Weight	0,039987	0,023069	0,000003	0,020453	0,039891	0,058751	0,125376	0,945593
Portugal Weight	0,040143	0,023080	0,000003	0,020577	0,040219	0,059103	0,113733	-0,005576
Russia Weight	0,039960	0,023021	0,000004	0,020400	0,039951	0,058649	0,124992	2,978830
South Africa Weight	0,039812	0,023094	0,000003	0,020214	0,039676	0,058754	0,129951	0,024114
South Korea Weight	0,040121	0,023099	0,000009	0,020547	0,040169	0,059105	0,113348	0,368079
Spain Weight	0,040162	0,023057	0,000000	0,020694	0,040094	0,058949	0,112293	-0,633277
Sweden Weight	0,040024	0,023193	0,000000	0,020361	0,039984	0,058950	0,117331	0,299641
Switzerland Weight	0,039866	0,023051	0,000001	0,020210	0,039735	0,058721	0,120694	-0,157755
UK Weight	0,040096	0,023118	0,000000	0,020350	0,040113	0,059014	0,136021	0,141864
US Weight	0,039977	0,023027	0,000000	0,020453	0,040043	0,058674	0,117503	0,063321

Table 2: The total returns with TVER

6.2 Constant Expected return method results

When we simulated the asset allocation with the constant expected return method we received the two plots you can see below. The efficient frontier is the set of optimal portfolios that offers the highest expected return for a level of risk, which is presented as the standard deviation. The problem is to find the split across the assets that achieve a target return whilst minimizing this variance of return. This is a standard optimization problem that can be answered by our Python program, which contains iterative search methods for optimization. We simulated 50.000 portfolios with different combinations of weights, that as a result generated different expected returns and expected volatility. Each point lying on the top of the green area represents an optimal combination of stocks, that maximizes the expected return for the given level of risk.

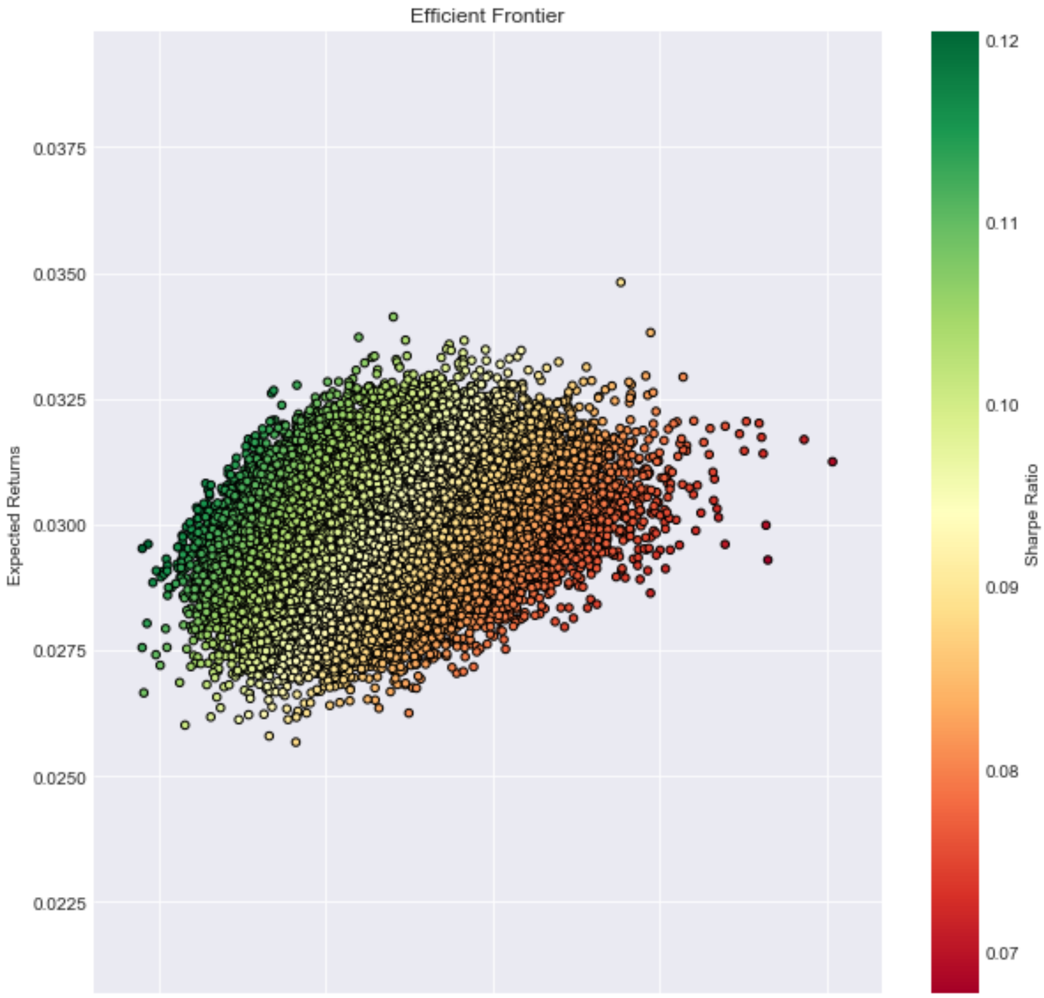

Figure 10: Efficient frontier for the CER portfolio

On the second efficient frontier, we have now added two marks. The blue mark is the minimum variance portfolio, while the red mark is the maximum Sharpe ratio portfolio. These two points give the highest possible return for each portfolio, with the lowest risk possible.

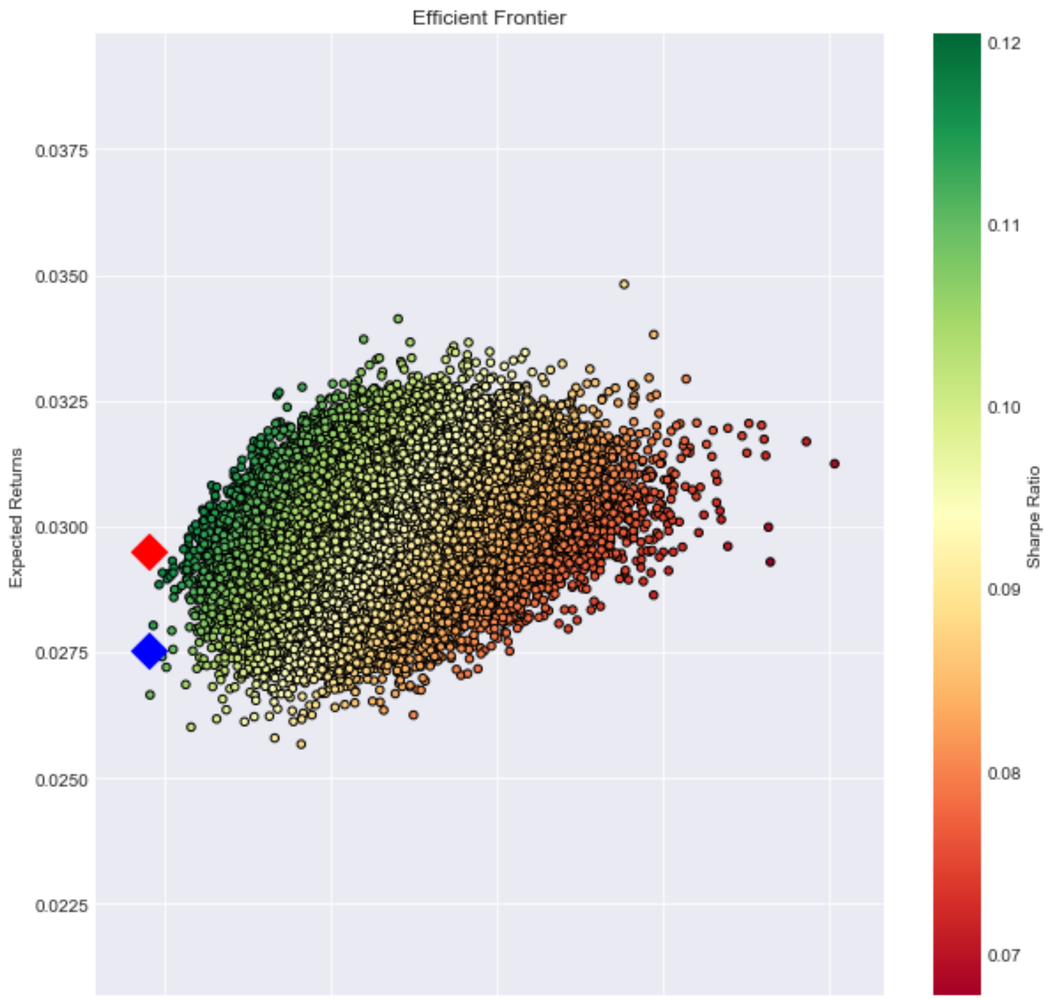

Figure 11: Efficient frontier for the CER portfolio with minimum variance portfolio (blue) and maximum Sharpe ratio portfolio (red).

The weights of the optimal portfolio with both minimum variance and Sharpe ratio are as follows:

CER: Minimum Variance Portfolio	Weights	CER: Sharpe Ratio Portfolio	Weights
Returns	2,75 %	Returns	2,95 %
Volatility	24,51 %	Volatility	24,52 %
Sharpe Ratio	11,24 %	Sharpe Ratio	12,04 %
Austria Weight	3,40 %	Austria Weight	2,94 %
Belgium Weight	6,42 %	Belgium Weight	7,16 %
Canada Weight	1,85 %	Canada Weight	5,42 %
Chech Republic Weight	7,28 %	Chech Republic Weight	6,47 %
Chile Weight	3,12 %	Chile Weight	5,56 %
China Weight	1,59 %	China Weight	6,92 %
Denmark Weight	1,87 %	Denmark Weight	2,30 %
France Weight	4,56 %	France Weight	8,01 %
Germany Weight	4,21 %	Germany Weight	1,64 %
Hungary Weight	2,90 %	Hungary Weight	1,36 %
Israel Weight	1,44 %	Israel Weight	5,51 %
Italy Weight	1,65 %	Italy Weight	1,36 %
Japan Weight	10,21 %	Japan Weight	5,44 %
Mexico Weight	9,82 %	Mexico Weight	6,95 %
Netherland Weight	1,28 %	Netherland Weight	6,19 %
Poland Weight	6,29 %	Poland Weight	2,60 %
Portugal Weight	0,74 %	Portugal Weight	2,88 %
Russia Weight	5,96 %	Russia Weight	4,44 %
South Africa Weight	1,96 %	South Africa Weight	1,12 %
South Korea Weight	7,62 %	South Korea Weight	1,16 %
Spain Weight	0,88 %	Spain Weight	0,91 %
Sweden Weight	0,23 %	Sweden Weight	1,19 %
Switzerland Weight	1,31 %	Switzerland Weight	1,37 %
UK Weight	3,45 %	UK Weight	3,17 %
US Weight	9,97 %	US Weight	7,92 %

Table 3: Country weights for CER

6.3 Time-Varying Expected Return method results

When we simulated the asset allocation with the time-varying expected return method we received the plots below. We used a rolling time window of three years to calculate the excess return. This simply means, that we conducted regressions over and over again to create "new" observations, using subsamples from our original dataset. We used Realized Rolling Sharpe ratios, which imply that we calculated the risk and return that would have resulted from using the investments weights recommended by the model.

As mentioned above, the efficient frontier is the set of optimal portfolios with the highest expected return for a level of risk. Here we also simulated 50.000 portfolios with different combinations of weights, and every point located on the outside of the green area is an optimal combination of stocks, that maximizes the expected return for the given level of risk.

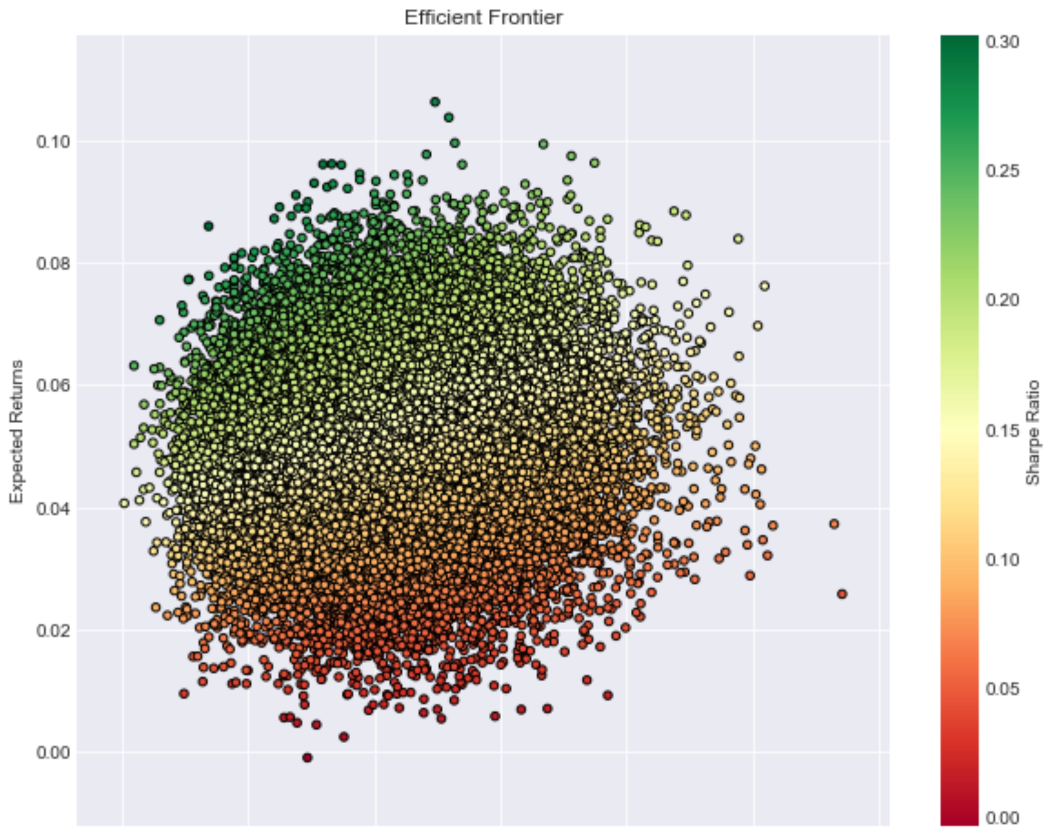

Figure 12: Efficient frontier for the TVER portfolio

On the second efficient frontier, we have now added two marks once more. The blue mark is the minimum variance portfolio, while the red mark is the maximum Sharpe ratio portfolio. These two points give the highest possible return for each portfolio, with the lowest risk possible.

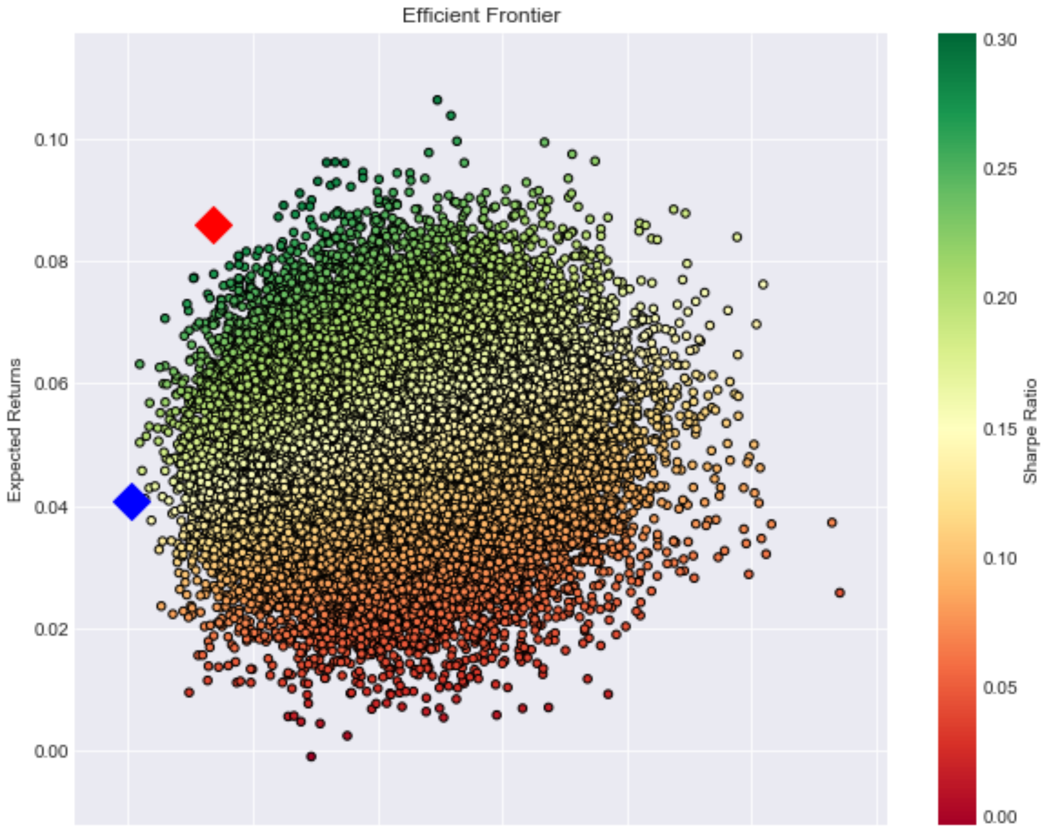

Figure 13: Efficient frontier for the TVER portfolio with minimum variance portfolio (blue) and maximum Sharpe ratio portfolio (red).

The results we achieved from the TVER method have some small differences from the CER method. The optimal portfolio with both minimum variance and Sharpe ratio is as follows:

TVER: Sharpe Ratio Portfolio	Weights
Returns	8,59 %
Volatility	28,42 %
Sharpe Ratio	30,23 %
Austria Weight	0,09 %
Belgium Weight	1,55 %
Canada Weight	8,02 %
Chech Republic Weight	6,31 %
Chile Weight	0,04 %
China Weight	2,32 %
Denmark Weight	6,92 %
France Weight	8,11 %
Germany Weight	1,51 %
Hungary Weight	3,66 %
Israel Weight	8,69 %
Italy Weight	0,82 %
Japan Weight	7,06 %
Mexico Weight	3,40 %
Netherland Weight	4,08 %
Poland Weight	5,72 %
Portugal Weight	0,91 %
Russia Weight	5,31 %
South Africa Weight	1,81 %
South Korea Weight	2,43 %
Spain Weight	6,78 %
Sweden Weight	2,89 %
Switzerland Weight	1,41 %
UK Weight	1,79 %
US Weight	8,38 %

TVER: Minimum Variance Portfoli	Weights
Returns	4,06 %
Volatility	25,08 %
Sharpe Ratio	16,20 %
Austria Weight	1,12 %
Belgium Weight	5,14 %
Canada Weight	4,72 %
Chech Republic Weight	3,87 %
Chile Weight	0,14 %
China Weight	4,13 %
Denmark Weight	3,94 %
France Weight	5,43 %
Germany Weight	2,06 %
Hungary Weight	1,16 %
Israel Weight	9,24 %
Italy Weight	5,30 %
Japan Weight	7,26 %
Mexico Weight	7,26 %
Netherland Weight	1,47 %
Poland Weight	5,48 %
Portugal Weight	1,13 %
Russia Weight	3,73 %
South Africa Weight	8,54 %
South Korea Weight	0,18 %
Spain Weight	0,25 %
Sweden Weight	8,47 %
Switzerland Weight	1,53 %
UK Weight	7,50 %
US Weight	0,60 %

Table 4: Country weights for TVER

6.4 Comparison with the benchmark

For comparison of models, we will analyze the two efficient frontier we have simulated, and especially look at the location of the tangency portfolio of the portfolios. Specifically, the portfolios that use the different forecasted returns, variances and correlations and compare this to a situation with full knowledge, in which we insert the realized returns and realized weights of the FTSE. In the table below, we have compared the different returns, volatility and Sharpe ratio for our models with the FTSE benchmark.

	Returns	Volatility	Sharpe Ratio
CER: Max Sharpe Ratio Portfolio	2,95 %	24,52 %	0,12
TVER: Max Sharpe Ratio Portfolio	8,59 %	28,42 %	0,30
CER: Minimum Variance Portfolio	2,75 %	24,51 %	0,11
TVER: Minimum Variance Portfolio	4,06 %	25,08 %	0,16
FTSE Benchmark Portfolio	11,60 %	8,90 %	0,50

Table 5: Comparison of the models return, volatility and Sharpe ratio with the FTSE benchmark portfolio

Table 6 shows the different weights for every simulated portfolio compared with the FTSE benchmark index. While figure 14 and 15 visualize and compare the weighting for both CER and TVER portfolio with the benchmark in a histogram.

Overview of all portfolio weightings	CER: Sharpe Ratio Portfolio	CER: Minimum Variance Portfolio	TVER: Sharpe Ratio Portfolio	TVER: Minimum Variance Portfolio	FSTE Benchmark Portfolio
Austria Weight	2,94 %	3,40 %	0,09 %	1,12 %	0,22 %
Belgium Weight	7,16 %	6,42 %	1,55 %	5,14 %	0,73 %
Canada Weight	5,42 %	1,85 %	8,02 %	4,72 %	2,11 %
Chech Republic Weight	6,47 %	7,28 %	6,31 %	3,87 %	0,01 %
Chile Weight	5,56 %	3,12 %	0,04 %	0,14 %	0,15 %
China Weight	6,92 %	1,59 %	2,32 %	4,13 %	3,46 %
Denmark Weight	2,30 %	1,87 %	6,92 %	3,94 %	1,05 %
France Weight	8,01 %	4,56 %	8,11 %	5,43 %	5,31 %
Germany Weight	1,64 %	4,21 %	1,51 %	2,06 %	5,51 %
Hungary Weight	1,36 %	2,90 %	3,66 %	1,16 %	0,04 %
Israel Weight	5,51 %	1,44 %	8,69 %	9,24 %	0,17 %
Italy Weight	1,36 %	1,65 %	0,82 %	5,30 %	1,69 %
Japan Weight	5,44 %	10,21 %	7,06 %	7,26 %	9,17 %
Mexico Weight	6,95 %	9,82 %	3,40 %	7,26 %	0,35 %
Netherland Weight	6,19 %	1,28 %	4,08 %	1,47 %	2,05 %
Poland Weight	2,60 %	6,29 %	5,72 %	5,48 %	0,14 %
Portugal Weight	2,88 %	0,74 %	0,91 %	1,13 %	0,12 %
Russia Weight	4,44 %	5,96 %	5,31 %	3,73 %	0,43 %
South Africa Weight	1,12 %	1,96 %	1,81 %	8,54 %	0,88 %
South Korea Weight	1,16 %	7,62 %	2,43 %	0,18 %	1,92 %
Spain Weight	0,91 %	0,88 %	6,78 %	0,25 %	1,87 %
Sweden Weight	1,19 %	0,23 %	2,89 %	8,47 %	1,74 %
Switzerland Weight	1,37 %	1,31 %	1,41 %	1,53 %	4,50 %
UK Weight	3,17 %	3,45 %	1,79 %	7,50 %	9,67 %
US Weight	7,92 %	9,97 %	8,38 %	0,60 %	36,01 %

Table 6: Comparison of the model's weights with the FTSE benchmark portfolio

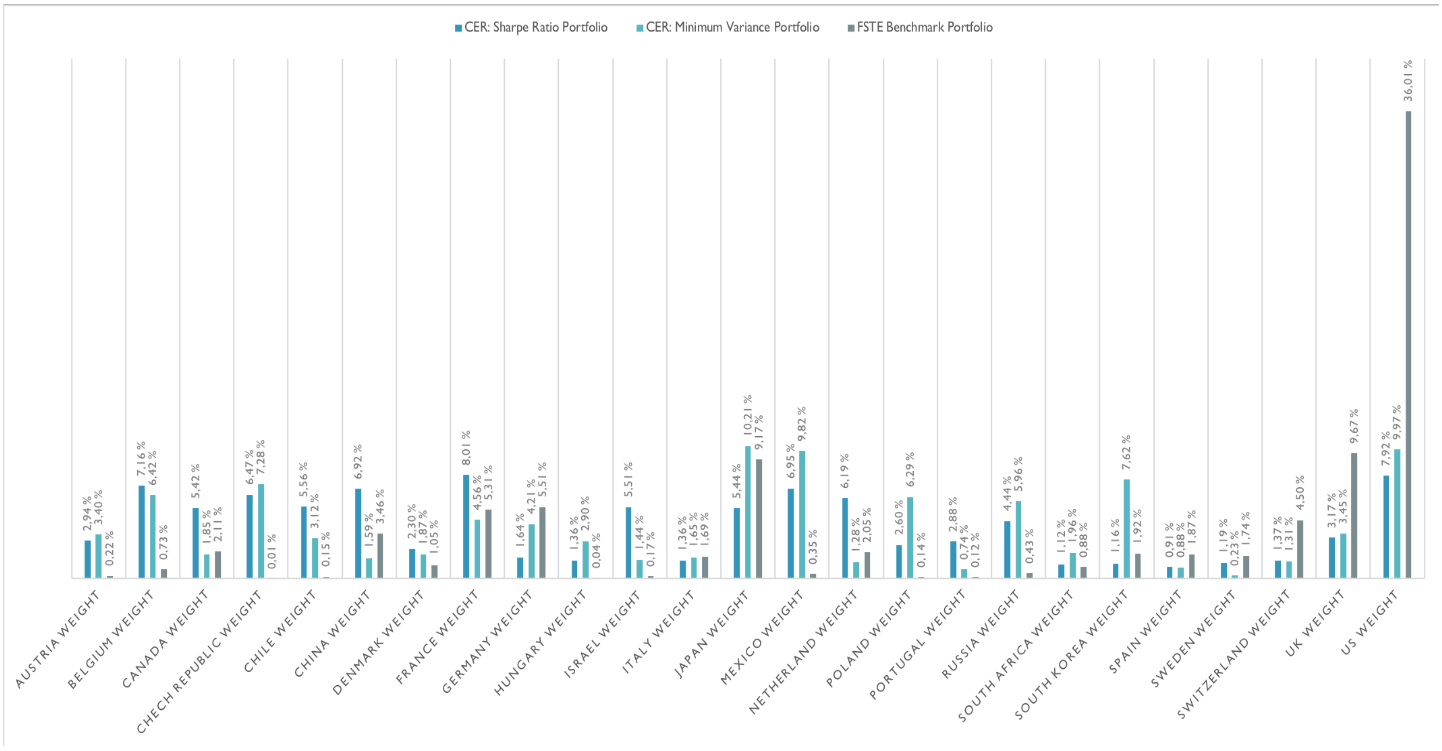

Figure 14: CER vs Benchmark weights

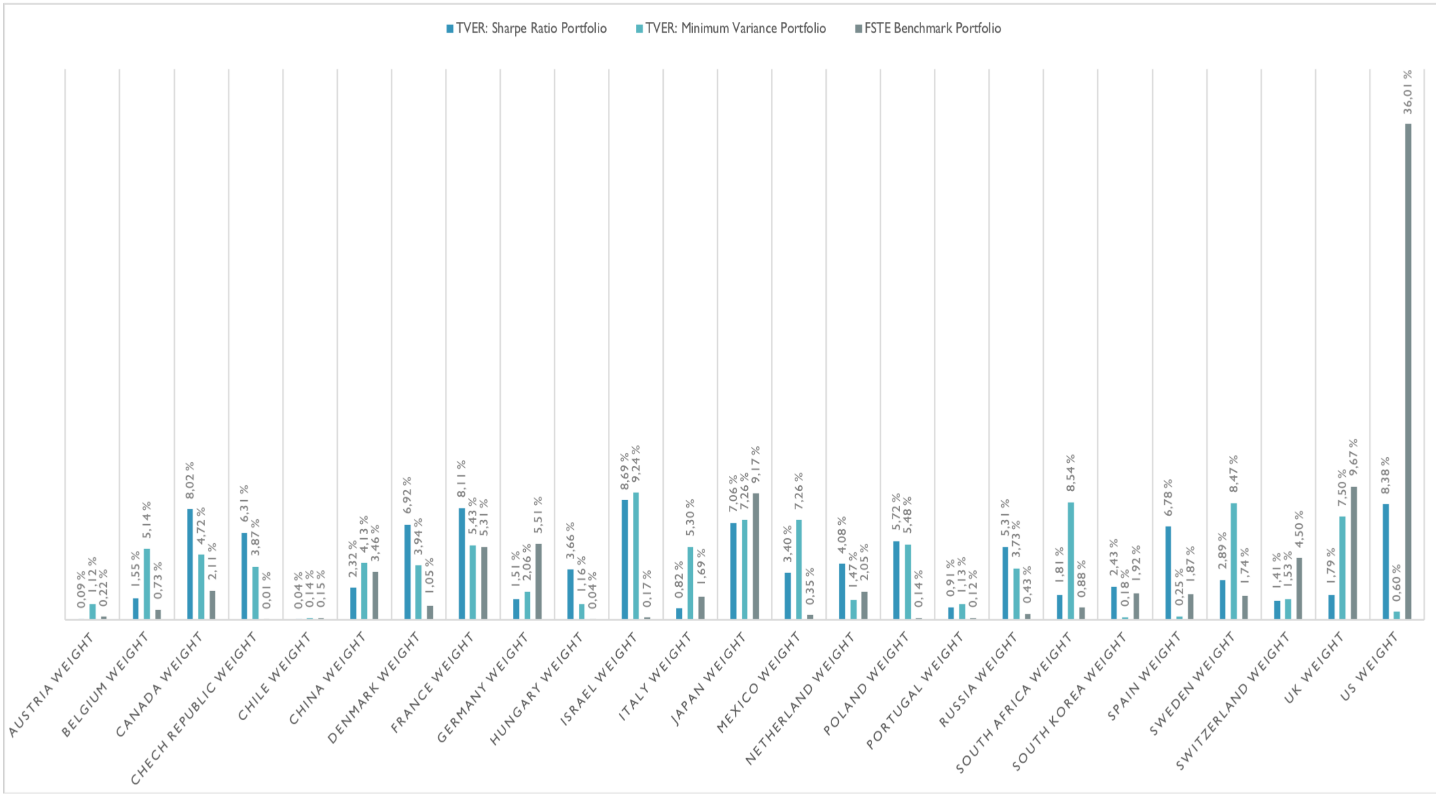

Figure 15: TVER vs Benchmark weights

6.5 Discussion

When we started this research, we wanted to examine if it is possible to optimize the asset allocation of the Norwegian Government Pension Fund – Global, without looking at influencing factors such as political, economic and ethical decisions, and rather concentrate on the risk and return relationship. After comparing the results from our estimation in Python with the realized data from the FTSE benchmark index, we see that the results are somewhat different.

As mentioned in chapter 3.3 there are essentially two types of investors, a risk-averse or a risk-loving investor. The risk-averse investor would construct the minimum variance portfolio. Which from our results, has an expected return of 2,75% and an accompanying expected volatility of 24,51% and a Sharp ratio of 0,11 for the CER portfolio, and the TVER portfolio has an expected return of 4,06% with an accompanying expected volatility of 25,08% and a Sharp ratio of 0,16. With the minimum variance portfolio, one will get the highest possible return, with the lowest level of risk. While an investor who is risk-neutral or a risk-lover, an investor seeking the maximum risk-adjusted return, would construct the maximum Sharpe ratio portfolio. This portfolio has an expected return of 2,95% and an accompanying expected volatility of 24,52% and a Sharp ratio of 0,12 for the CER portfolio, and the TVER portfolio has an expected return of 8,59% with an accompanying expected volatility of 28,42% and a Sharp ratio of 0,30. If an NBIM investor chooses the latter portfolio, then the investor is willing to take on more risk to achieve a higher return.

Our results show a sizeable difference between the constant- and time-varying expected return results. This is as expected based on the literature. The assumption of constant risk premia implies that the composition of an optimal portfolio is constant over time for both short-term and long-term investors. However, much research suggests that expected asset returns seem to vary so that investment opportunities are not constant (Norges Bank Investment Management, 2012). The fact that risk premia is time-varying, generates time variation in optimal portfolios. Both short-term and long-term investors should seek to "time the markets", holding more risky assets when the rewards for doing so are high. Campbell and Viceira (1999) show that there are large utility losses from holding a constant-mix portfolio when risk premia are time-varying. In other words, the simulation of the time-varying portfolio should be closer to reality.

Further, when we look at the FTSE benchmark index it shows a result of 11,60% return with an accompanying volatility of 8,90% and a Sharpe ratio of 0,50, which is far-off from most of our simulated portfolios. However, the time-varying maximum Sharpe ratio portfolio has a slightly better return of 8,59%, but a higher volatility of 28,42%, which is a Sharpe ratio which is not far from the benchmark. After all, the benchmark includes more countries than we have in our research, suggesting that the index is more diversified than our portfolio. We are missing some countries in South America, the Middle East and Asia that may have an impact because many of them are emerging economies. Moreover, we have a time frame of ten years, which can be considered as somewhat short in a stock marked context. Regarding this, our results are not that far-off considering that we also do not take any political, economic or ethical decisions in our investment strategy.

Nevertheless, our portfolios have quite different weights than the FTSE benchmark index as you can see in figure 16. In this figure we have plotted the weights from the time-varying maximum Sharpe ratio portfolio and the weights from the benchmark. The biggest and most secure countries like Germany, France, United States and United Kingdom and emerging countries like China obtains the highest weights in the benchmark. The time-varying maximum Sharpe ratio portfolio invests in these countries as well, though the weights are more evenly distributed between all the countries than in the benchmark index. Our portfolio has higher weights in emerging countries like Russia, Poland and the Czech Republic which will naturally give the portfolio higher volatility because there is higher risk involved with these investments. It is also interesting to notice that our portfolios chose such a low weight in the United States compared with the benchmark, that might have something to do with us not looking at political or economic influencing factors.

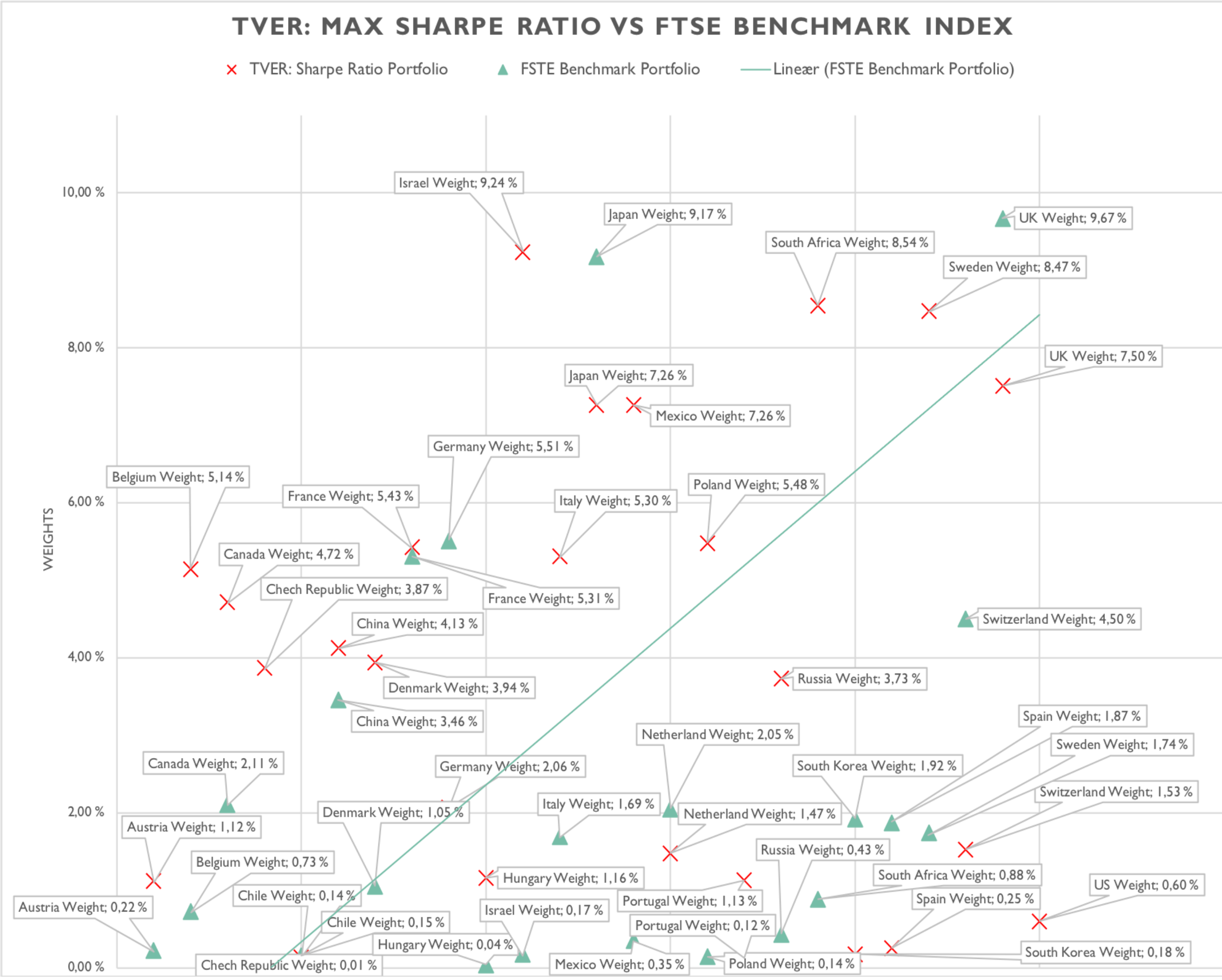

Figure 16: Comparison of the TVER: sharpe ratio weights with the FTSE Benchmark Index

7. Conclusion and limitations

Modern portfolio theory implies that there are benefits from investing in different countries to diversify and lower the overall risk. As the main purpose of this study – we wanted to examine if the optimal asset allocation for the Norwegian Government Pension Fund – Global can be improved without taking any political, economic or ethical assumptions.

To test this, we programmed a mean-variance optimization model in Python, where the highest Sharpe ratio and the minimum variance was used as measurements for performance. Mean-variance optimization focuses on the first two moments of a distribution which assumes that the returns are normally distributed. We used excess return and standard deviation for all countries to calculate a constant expected return portfolio, and a rolling window of three years to simulate the time-varying expected return portfolio. Further we divided these two portfolios into a maximum Sharpe ratio portfolio and a minimum variance portfolio. The minimum variance portfolio is seeking the lowest volatility of return, while the maximum Sharpe ratio portfolio seeks the maximum risk-adjusted return. Our computations gave us in the end four different portfolios with different optimal asset allocations.

From our Python program and computations, the time-varying expected return Sharpe ratio portfolio has the highest Sharpe ratio and is the closest to the benchmark. This is simulated without having any political, economic or ethical decisions in the investment strategy. Our weights are more evenly diversified throughout the countries than the benchmark index and is targeting more emerging markets. As a result of this, our portfolio has higher volatility too. NBIM are investing more heavily in secure countries, where one is almost guaranteed a high return without unnecessary risk. We believe our research proves that it is possible to achieve a decent return by only concentrating on the return and variance trade-off. This could be of value as a correction to a more complex model – as increased complexity always introduces and increased chance for introducing errors. In the future this could be a possible investment strategy for NBIM if they are willing to take on more risk. Even though, our portfolios were not too far below the benchmark. We conclude that we were not able to outperform the FTSE benchmark with our asset allocation by only looking at the risk-return relationship, and that even better results can be obtained by including other influencing factors such as political, economic and ethical decisions in the investment strategies.

7.1 Limitations and recommendations for future research

This study is subject to certain limitations regarding the number of the countries included. NBIM tend to disclose very limited public information. Therefore, there is not a well-designed and advanced database or other sources that give proper and complete data about the Norwegian Government Pension Fund's transactions. Consequently, we have worked with 25 countries and their historical return data. Without doubt, if we had had more observations, our results would have been more reliable. We also have a somewhat limited time period of ten years, which is considered short in a stock market context, however it is considered a suited time period for a master thesis.

Further studies could include more assets classes in the optimization model such as fixed income and real estate. This study limited the scope to only equity and 25 countries. Additional research could analyze more in-depth when it comes to the influence of political-, economic- and ethical decisions and review other risk-factors. An analysis of individual markets can also be done, rather than investing in a benchmark index. Currency hedging and short selling were not addressed in this study and therefore, further analysis on asset allocation could factor in these two elements.

8. References

Allayannis, G., & Weston, J. P. (2001). The Use of Foreign Currency Derivatives and Firm Market Value. *Review of Financial Studies, 14*(1), pp. 243-76.

Arrow, K. J. (1971). *Essays in the theory of risk-bearing* (Vol. 80). Amsterdam: North-Holland.

Becker, C., Ferson, W., Myers, D. H., & Schil, M. J. (199). Conditional market timing with benchmark investors. *Journal of Financial Economics, 52*, pp. 119-148.

Black, F. (1972). Capital Market Equilibrium with Restricted Borrowing. *The Journal of Business, 45*(3), pp. 444-455.

Bode Z., K. A. (2003). *Essentials of investment.* McGraw-Hill.

Bodie, Z., Kane, A., & Marcus, A. J. (2012). *Essentials of Investments* (9th ed.). McGraw-Hill / Irwin.

Bodnar, G., Tang, C., & Weintrop, J. (1999). Both Sides of Corporate Diversification: The Value Impacts of Geographic and Industrial Diversification. John Hopkins University, Washington and Beruch College, New York, USA.

Brentani, C. (2004). *Portfolio Management in Practice.* Oxford: Elsevier Butterwoth-Heinemann.

Campbell, J. Y., & Viceira, L. (1999). Consumption and portfolio decisions when expected returns are time varying. *Quarterly Journal of Economics*, pp. 433-495.

Charhart, M. M. (1997). On Persistence in Mutual Fund Performance. *The Journal of Finance, 52*(1), pp. 57-82.

Chen, N.-f., Roll, R., & Ross, S. (1986). Economic Forces and the Stock Market. *The Journal of Business, 59*(3), pp. 383-403.

Cochrane, J. H. (2011). Presidential Address: Discount Rates. *The Journal of Finance, 66*(4), pp. 1047-1108.

Cox, J. C. (1976, May). A Survey of some New Results in Financial Option Pricing Theory . *Journal of Finance, 31*(2), pp. 383-402.

Daniel, K., Grinblatt, M., Titman, S., & Wermers, R. (1997). Measuring Mutual Fund Performance with Characteristic-Based Benchmarks. *The Journal of Finance, 52*(3), pp. 1035-1058.

Fama, E. F. (1991). Efficient Capital Markets: II. *The Journal of Finance, 46*(5), pp. 1575-1617.

Fama, E. F., & French, K. R. (1993). Common risk factors in the returns on stocks and bonds. *Journal of Financial Economics, 33*(1), pp. 3-56.

Favero, C. A., & Guidolin, M. (2017, August 27). *The Econometrics of Asset Management: An Introduction.* Retrieved from Bocconi University: http://didattica.unibocconi.eu/myigier/index.php?IdUte=48917&idr=3008&lingua=eng&comando=Apri

French, K. S., & Stambaugh, R. (1987). Expected stock returns and volatility. *Journal of Financial Economics*(19(1)), pp. 3-29.

Green, R. C., & Hollifield, B. (1992). When Will Mean-Variance Efficient Portfolios be Well Diversified? *The Journal of Finance, 47*(5), pp. 1785-1809.

Harding, D. (2002). *A Critique of the Sharpe Ratio.* White paper, Winton Capital Management .

Henriksson, R. D., & Merton, R. C. (1981). On the Market Timing and Investment Performance of Managed Portfolios II - Statistical Procedures for Evaluating Forecasting Skills. *54*, pp. 513-533.

Hudson-Wilson, S. (1990). New Trends in Portfolio Theory. *Journal of Property Management*(55), p. 57.

Jehle, G. A. (2000). *Advanced microeconomic theory* . Addison Wesley: number 2.

Jensen, M. C. (1978). Some Anomalous Evidence Regarding Market Efficiency. *Journal of Financial Economics, 6*, pp. 95-101.

Kent, D., Grinblatt, M., Titman, S., & Wermers, R. S. (1997). Measuring Mutual Fund Performance with Characteristic-Based Benchmarks. *Journal of Finance, 52*, pp. 1035-1057.

Lakonishok, J., Shleifer, A., & Vishny, R. W. (1994). Contrarian Investment, Extrapolation, and Risk. *The Journal of FInance, 49*(5), pp. 1541-1578.

Lintner, J. (1965). The Valuation of Risk Assets and the Selection of Risky Investments in Stock Portfolios and Capital Budgets. *The Review of Economics and Statistics, 47*(1), pp. 13-37.

Malkiel, B. G. (2003). The Efficient Market Hypothesis and Its Critics. *Journal of Economic Perspectives, 17*(1), pp. 59-82.

Markowitz, H. (1952). Portfolio Selection. *The Journal of Finance, 7*(1), pp. 77-91.

Morck, R., & Yeung, B. (1991). Why Investors Value Multinationality. *The Journal of Business, 64*(2), pp. 165-87.

Norges Bank Investment Management. (2012, 03 30). *NBIM - Discussion Note.* Retrieved from https://www.nbim.no/globalassets/documents/dicussion-paper/2012/discussionnote_3-12-final.pdf

Norges Bank Investment Management. (2016). *Return and Risk - Government Pension Fund Global.* Norges Bank Investment Management.

Norges Bank Investment Management. (2017). *Government Pension Fund Global - Annual Report 2016.* Norges Bank Investment Management.

Norwegian Ministry of Finance. (2017). *Meld. St. 26 (2016–2017) Report to the Storting - The Management of the Government Pension Fund in 2016.* Norwegian Ministry of Finance.

Pratt, J. W. (n.d.). Risk-Aversion in the Small and in the Large . *Econometrica 32*, pp. 122-136.

Sharpe, W. (1964). Capital Asset Prices: A theory of market equilibrium under conditions of risk. *The Journal of Finance, 19*(3), pp. 425-442.

Sharpe, W. F. (1966). Mutual Fund Performance. *The Journal of Business*(39(1)), pp. 119-138.

Sharpe, W. F. (1994). The Sharpe Ratio. *The Journal of Portfolio Management*(21(1)), pp. 49-58.

Statman, M. (1987). How many stocks make a diversified portfolio? *Journal of Financial and Quantitative Analysis*(22), pp. 353-364.

Tobin, J. (1958). Liquidity Preference as Behavior towards Risk. *The Review of Economic Studies, 25*(2), pp. 65-86.

Appendix

Master Thesis Code

```
In [192]:
import numpy as np
import pandas as pd
from IPython.display import Image
import matplotlib.pyplot as plt
import cvxopt as opt
from cvxopt import blas, solvers
import seaborn as sns
import scipy.optimize as sco
import statsmodels.tsa.api as smt
from statsmodels.tsa.stattools import adfuller, kpss
%matplotlib inline
Import data
In [1]:
data = pd.read_excel('MASTER.xlsx', "Monthly",index_col = 'Date')
data.columns = data.columns.str.lower()
data.head()
Building returns
Norway
In [194]:
norway = data.copy()
norway = norway [['norwayltgovbondyield10y','norway3m','norwaycpi','norwaydivyld']]
#norway risk-free
norway ['RiskFree'] = np.log(1+(norway['norway3m']).shift(1)/(100*12))
norwayRiskFree = norway['RiskFree']
Australia
In [ ]:
australia = data.copy()
australia = australia[['australialtgovbondyield10y', 'australia3m', 'australiacpi', 'australiadivyld']]
australia.head()
#Monthly log dividend yield
australia['australiaStock_DY'] = australia['australiadivyld']
australia['australiaLog_Stock_DY'] = np.log(australia['australiaStock_DY'])
#Monthly lagged stock index
australia['australiaStock_Index'] = australia['australiacpi']
australia['australialag_Stock_Index'] = (australia['australiaStock_Index']).shift(1)
lag_Stock_Index = australia['australialag_Stock_Index']
#Stock Return
australia['australiaStock_Ret'] = np.log(australia['australiaStock_Index']/australia['australialag_Stock_Index']+australia['australiaStock_DY'])
#Stock Excess Return
australia['australiaStock_ExRet'] = australia['australiaStock_Ret'] - norway['RiskFree']
```

```
australia1 = australia[['australiaLog_Stock_DY','australialag_Stock_Index','australiaStock_Ret','australiaStock_ExRet']]
australia1 = australia1[~np.isnan(lag_Stock_Index)]
australia1.head()
Austria
In [ ]:

austria = data.copy()
austria = austria[['austrialtgovbondyield10y', 'austria3m', 'austriacpi', 'austriadivyld']]
austria.head()

#Monthly log dividend yield
austria['austriaStock_DY'] = austria['austriadivyld']
austria['austriaLog_Stock_DY'] = np.log(austria['austriaStock_DY'])

#Monthly lagged stock index
austria['austriaStock_Index'] = austria['austriacpi']
austria['austrialag_Stock_Index'] = (austria['austriaStock_Index']).shift(1)
lag_Stock_Index = austria['austrialag_Stock_Index']

#Stock Return
austria['austriaStock_Ret'] = np.log(austria['austriaStock_Index']/austria['austrialag_Stock_Index']+austria['austriaStock_DY'])

#Stock Excess Return
austria['austriaStock_ExRet'] = austria['austriaStock_Ret'] - norway['RiskFree']

austria1 = austria[['austriaLog_Stock_DY','austrialag_Stock_Index','austriaStock_Ret','austriaStock_ExRet']]
austria1 = austria1[~np.isnan(lag_Stock_Index)]
austria1.head()
Belgium
In [ ]:

belgium = data.copy()
belgium = belgium[['belgiumltgovbondyield10y', 'belgium3m', 'belgiumcpi', 'belgiumdivyld']]
belgium.head()

#Monthly log dividend yield
belgium['belgiumStock_DY'] = belgium['belgiumdivyld']
belgium['belgiumLog_Stock_DY'] = np.log(belgium['belgiumStock_DY'])

#Monthly lagged stock index
belgium['belgiumStock_Index'] = belgium['belgiumcpi']
belgium['belgiumlag_Stock_Index'] = (belgium['belgiumStock_Index']).shift(1)
lag_Stock_Index = belgium['belgiumlag_Stock_Index']

#Stock Return
belgium['belgiumStock_Ret'] = np.log(belgium['belgiumStock_Index']/belgium['belgiumlag_Stock_Index']+belgium['belgiumStock_DY'])

#Stock Excess Return
belgium['belgiumStock_ExRet'] = belgium['belgiumStock_Ret'] - norway['RiskFree']

belgium1 = belgium[['belgiumLog_Stock_DY','belgiumlag_Stock_Index','belgiumStock_Ret','belgiumStock_ExRet']]
belgium1 = belgium1[~np.isnan(lag_Stock_Index)]
belgium1.head()
Canada
```

```
In [ ]:

canada = data.copy()
canada = canada[['canadaltgovbondyield10y', 'canada3m', 'canadacpi', 'canadadivyld']]
canada.head()

#Monthly log dividend yield
canada['canadaStock_DY'] = canada['canadadivyld']
canada['canadaLog_Stock_DY'] = np.log(canada['canadaStock_DY'])

#Monthly lagged stock index
canada['canadaStock_Index'] = canada['canadacpi']
canada['canadalag_Stock_Index'] = (canada['canadaStock_Index']).shift(1)
lag_Stock_Index = canada['canadalag_Stock_Index']

#Stock Return
canada['canadaStock_Ret'] = np.log(canada['canadaStock_Index']/canada['canadalag_Stock_Index']+canada['canadaStock_DY'])


#Stock Excess Return
canada['canadaStock_ExRet'] = canada['canadaStock_Ret'] - norway['RiskFree']

canada1 = canada[['canadaLog_Stock_DY','canadalag_Stock_Index','canadaStock_Ret','canadaStock_ExRet']]
canada1 = canada1[~np.isnan(lag_Stock_Index)]
canada1.head()
Chech Republic
In [ ]:

chechrepublic = data.copy()
chechrepublic = chechrepublic[['chechrepublicltgovbondyield10y', 'chechrepublic3m', 'chechrepubliccpi', 'chechrepublicdivyld']]
chechrepublic.head()

#Monthly log dividend yield
chechrepublic['chechrepublicStock_DY'] = chechrepublic['chechrepublicdivyld']
chechrepublic['chechrepublicLog_Stock_DY'] = np.log(chechrepublic['chechrepublicStock_DY'])

#Monthly lagged stock index
chechrepublic['chechrepublicStock_Index'] = chechrepublic['chechrepubliccpi']
chechrepublic['chechrepubliclag_Stock_Index'] = (chechrepublic['chechrepublicStock_Index']).shift(1)
lag_Stock_Index = chechrepublic['chechrepubliclag_Stock_Index']

#Stock Return
chechrepublic['chechrepublicStock_Ret'] = np.log(chechrepublic['chechrepublicStock_Index']/chechrepublic['chechrepubliclag_Stock_Index']+chechrepublic['chechrepublicStock_DY'])


#Stock Excess Return
chechrepublic['chechrepublicStock_ExRet'] = chechrepublic['chechrepublicStock_Ret'] - norway['RiskFree']

chechrepublic1 = chechrepublic[['chechrepublicLog_Stock_DY','chechrepubliclag_Stock_Index','chechrepublicStock_Ret','chechrepublicStock_ExRet']]
chechrepublic1 = chechrepublic1[~np.isnan(lag_Stock_Index)]
chechrepublic1.head()
Chile
```

```
In [ ]:

chile = data.copy()
chile = chile[['chileltgovbondyield10y', 'chile3m', 'chilecpi', 'chiledivyld']]
chile.head()

#Monthly log dividend yield
chile['chileStock_DY'] = chile['chiledivyld']
chile['chileLog_Stock_DY'] = np.log(chile['chileStock_DY'])

#Monthly lagged stock index
chile['chileStock_Index'] = chile['chilecpi']
chile['chilelag_Stock_Index'] = (chile['chileStock_Index']).shift(1)
lag_Stock_Index = chile['chilelag_Stock_Index']

#Stock Return
chile['chileStock_Ret'] = np.log(chile['chileStock_Index']/chile['chilelag_Stock_Index']+chile['chileStock_DY'])


#Stock Excess Return
chile['chileStock_ExRet'] = chile['chileStock_Ret'] - norway['RiskFree']

chile1 = chile[['chileLog_Stock_DY','chilelag_Stock_Index','chileStock_Ret','chileStock_ExRet']]
chile1 = chile1[~np.isnan(lag_Stock_Index)]
chile1.head()
China
In [ ]:

china = data.copy()
china = china[['chinaltgovbondyield10y', 'china3m', 'chinacpi', 'chinadivyld']]
china.head()

#Monthly log dividend yield
china['chinaStock_DY'] = china['chinadivyld']
china['chinaLog_Stock_DY'] = np.log(china['chinaStock_DY'])

#Monthly lagged stock index
china['chinaStock_Index'] = china['chinacpi']
china['chinalag_Stock_Index'] = (china['chinaStock_Index']).shift(1)
lag_Stock_Index = china['chinalag_Stock_Index']

#Stock Return
china['chinaStock_Ret'] = np.log(china['chinaStock_Index']/china['chinalag_Stock_Index']+china['chinaStock_DY'])


#Stock Excess Return
china['chinaStock_ExRet'] = china['chinaStock_Ret'] - norway['RiskFree']

china1 = china[['chinaLog_Stock_DY','chinalag_Stock_Index','chinaStock_Ret','chinaStock_ExRet']]
china1 = china1[~np.isnan(lag_Stock_Index)]
china1.head()
Denmark
In [ ]:

denmark = data.copy()
denmark = denmark[['denmarkltgovbondyield10y', 'denmark3m', 'denmarkcpi', 'denmarkdivyld']]
```

```
denmark.head()

#Monthly log dividend yield
denmark['denmarkStock_DY'] = denmark['denmarkdivyld']
denmark['denmarkLog_Stock_DY'] = np.log(denmark['denmarkStock_DY'])

#Monthly lagged stock index
denmark['denmarkStock_Index'] = denmark['denmarkcpi']
denmark['denmarklag_Stock_Index'] = (denmark['denmarkStock_Index']).shift(1)
lag_Stock_Index = denmark['denmarklag_Stock_Index']

#Stock Return
denmark['denmarkStock_Ret'] = np.log(denmark['denmarkStock_Index']/denmark['denmarklag_Stock_Index']+denmark['denmarkStock_DY'])

#Stock Excess Return
denmark['denmarkStock_ExRet'] = denmark['denmarkStock_Ret'] - norway['RiskFree']

denmark1 = denmark[['denmarkLog_Stock_DY','denmarklag_Stock_Index','denmarkStock_Ret','denmarkStock_ExRet']]
denmark1 = denmark1[~np.isnan(lag_Stock_Index)]
denmark1.head()
France
In [ ]:

france = data.copy()
france = france[['franceltgovbondyield10y', 'france3m', 'francecpi', 'francedivyld']]
france.head()

#Monthly log dividend yield
france['franceStock_DY'] = france['francedivyld']
france['franceLog_Stock_DY'] = np.log(france['franceStock_DY'])

#Monthly lagged stock index
france['franceStock_Index'] = france['francecpi']
france['francelag_Stock_Index'] = (france['franceStock_Index']).shift(1)
lag_Stock_Index = france['francelag_Stock_Index']

#Stock Return
france['franceStock_Ret'] = np.log(france['franceStock_Index']/france['francelag_Stock_Index']+france['franceStock_DY'])

#Stock Excess Return
france['franceStock_ExRet'] = france['franceStock_Ret'] - norway['RiskFree']

france1 = france[['franceLog_Stock_DY','francelag_Stock_Index','franceStock_Ret','franceStock_ExRet']]
france1 = france1[~np.isnan(lag_Stock_Index)]
france1.head()
Germany
In [ ]:

germany = data.copy()
germany = germany[['germanyltgovbondyield10y', 'germany3m', 'germanycpi', 'germanydivyld']]
germany.head()

#Monthly log dividend yield
germany['germanyStock_DY'] = germany['germanydivyld']
```

```
    germany['germanyLog_Stock_DY'] = np.log(germany['germanyStock_DY'])

    #Monthly lagged stock index
    germany['germanyStock_Index'] = germany['germanycpi']
    germany['germanylag_Stock_Index'] = (germany['germanyStock_Index']).shift(1)
    lag_Stock_Index = germany['germanylag_Stock_Index']

    #Stock Return
    germany['germanyStock_Ret'] = np.log(germany['germanyStock_Index']/germany['germanylag_Stock_Index']+germany['germanyStock_DY'])

    #Stock Excess Return
    germany['germanyStock_ExRet'] = germany['germanyStock_Ret'] - norway['RiskFree']

    germany1 = germany[['germanyLog_Stock_DY','germanylag_Stock_Index','germanyStock_Ret','germanyStock_ExRet']]
    germany1 = germany1[~np.isnan(lag_Stock_Index)]
    germany1.head()
    Hungary
    In [ ]:

    hungary = data.copy()
    hungary = hungary[['hungaryltgovbondyield10y', 'hungary3m', 'hungarycpi', 'hungarydivyld']]
    hungary.head()

    #Monthly log dividend yield
    hungary['hungaryStock_DY'] = hungary['hungarydivyld']
    hungary['hungaryLog_Stock_DY'] = np.log(hungary['hungaryStock_DY'])

    #Monthly lagged stock index
    hungary['hungaryStock_Index'] = hungary['hungarycpi']
    hungary['hungarylag_Stock_Index'] = (hungary['hungaryStock_Index']).shift(1)
    lag_Stock_Index = hungary['hungarylag_Stock_Index']

    #Stock Return
    hungary['hungaryStock_Ret'] = np.log(hungary['hungaryStock_Index']/hungary['hungarylag_Stock_Index']+hungary['hungaryStock_DY'])

    #Stock Excess Return
    hungary['hungaryStock_ExRet'] = hungary['hungaryStock_Ret'] - norway['RiskFree']

    hungary1 = hungary[['hungaryLog_Stock_DY','hungarylag_Stock_Index','hungaryStock_Ret','hungaryStock_ExRet']]
    hungary1 = hungary1[~np.isnan(lag_Stock_Index)]
    hungary1.head()
    Israel
    In [ ]:

    israel = data.copy()
    israel = israel[['israelltgovbondyield10y', 'israel3m', 'israelcpi', 'israeldivyld']]
    israel.head()

    #Monthly log dividend yield
    israel['israelStock_DY'] = israel['israeldivyld']
    israel['israelLog_Stock_DY'] = np.log(israel['israelStock_DY'])

    #Monthly lagged stock index
```

```
israel['israelStock_Index'] = israel['israelcpi']
israel['israellag_Stock_Index'] = (israel['israelStock_Index']).shift(1)
lag_Stock_Index = israel['israellag_Stock_Index']

#Stock Return
israel['israelStock_Ret'] = np.log(israel['israelStock_Index']/israel['israellag_Stock_Index']+israel['israelStock_DY'])


#Stock Excess Return
israel['israelStock_ExRet'] = israel['israelStock_Ret'] - norway['RiskFree']

israel1 = israel[['israelLog_Stock_DY','israellag_Stock_Index','israelStock_Ret','israelStock_ExRet']]
israel1 = israel1[~np.isnan(lag_Stock_Index)]
israel1.head()
Italy
In [ ]:

italy = data.copy()
italy = italy[['italyltgovbondyield10y', 'italy3m', 'italycpi', 'italydivyld']]
italy.head()

#Monthly log dividend yield
italy['italyStock_DY'] = italy['italydivyld']
italy['italyLog_Stock_DY'] = np.log(italy['italyStock_DY'])

#Monthly lagged stock index
italy['italyStock_Index'] = italy['italycpi']
italy['italylag_Stock_Index'] = (italy['italyStock_Index']).shift(1)
lag_Stock_Index = italy['italylag_Stock_Index']

#Stock Return
italy['italyStock_Ret'] = np.log(italy['italyStock_Index']/italy['italylag_Stock_Index']+italy['italyStock_DY'])


#Stock Excess Return
italy['italyStock_ExRet'] = italy['italyStock_Ret'] - norway['RiskFree']

italy1 = italy[['italyLog_Stock_DY','italylag_Stock_Index','italyStock_Ret','italyStock_ExRet']]
italy1 = italy1[~np.isnan(lag_Stock_Index)]
italy1.head()
Japan
In [ ]:

japan = data.copy()
japan = japan[['japanltgovbondyield10y', 'japan3m', 'japancpi', 'japandivyld']]
japan.head()

#Monthly log dividend yield
japan['japanStock_DY'] = japan['japandivyld']
japan['japanLog_Stock_DY'] = np.log(japan['japanStock_DY'])

#Monthly lagged stock index
japan['japanStock_Index'] = japan['japancpi']
japan['japanlag_Stock_Index'] = (japan['japanStock_Index']).shift(1)
lag_Stock_Index = japan['japanlag_Stock_Index']

```

```
    #Stock Return
    japan['japanStock_Ret'] = np.log(japan['japanStock_Index']/japan['japanlag_Stock_Index']+japan['japanStock_DY'])

    #Stock Excess Return
    japan['japanStock_ExRet'] = japan['japanStock_Ret'] - norway['RiskFree']

    japan1 = japan[['japanLog_Stock_DY','japanlag_Stock_Index','japanStock_Ret','japanStock_ExRet']]
    japan1 = japan1[~np.isnan(lag_Stock_Index)]
    japan1.head()
    Mexico
    In [ ]:

    mexico = data.copy()
    mexico = mexico[['mexicoltgovbondyield10y', 'mexico3m', 'mexicocpi', 'mexicodivyld']]
    mexico.head()

    #Monthly log dividend yield
    mexico['mexicoStock_DY'] = mexico['mexicodivyld']
    mexico['mexicoLog_Stock_DY'] = np.log(mexico['mexicoStock_DY'])

    #Monthly lagged stock index
    mexico['mexicoStock_Index'] = mexico['mexicocpi']
    mexico['mexicolag_Stock_Index'] = (mexico['mexicoStock_Index']).shift(1)
    lag_Stock_Index = mexico['mexicolag_Stock_Index']

    #Stock Return
    mexico['mexicoStock_Ret'] = np.log(mexico['mexicoStock_Index']/mexico['mexicolag_Stock_Index']+mexico['mexicoStock_DY'])

    #Stock Excess Return
    mexico['mexicoStock_ExRet'] = mexico['mexicoStock_Ret'] - norway['RiskFree']

    mexico1 = mexico[['mexicoLog_Stock_DY','mexicolag_Stock_Index','mexicoStock_Ret','mexicoStock_ExRet']]
    mexico1 = mexico1[~np.isnan(lag_Stock_Index)]
    mexico1.head()
    Netherland
    In [ ]:

    netherland = data.copy()
    netherland = netherland[['netherlandltgovbondyield10y', 'netherland3m', 'netherlandcpi', 'netherlanddivyld']]
    netherland.head()

    #Monthly log dividend yield
    netherland['netherlandStock_DY'] = netherland['netherlanddivyld']
    netherland['netherlandLog_Stock_DY'] = np.log(netherland['netherlandStock_DY'])

    #Monthly lagged stock index
    netherland['netherlandStock_Index'] = netherland['netherlandcpi']
    netherland['netherlandlag_Stock_Index'] = (netherland['netherlandStock_Index']).shift(1)
    lag_Stock_Index = netherland['netherlandlag_Stock_Index']

    #Stock Return
```

```
netherland['netherlandStock_Ret'] = np.log(netherland['netherlandStock_Index']/netherland['netherlandlag_Stock_Index']+netherland['netherlandStock_DY'])
#Stock Excess Return
netherland['netherlandStock_ExRet'] = netherland['netherlandStock_Ret'] - norway['RiskFree']
netherland1 = netherland[['netherlandLog_Stock_DY','netherlandlag_Stock_Index','netherlandStock_Ret','netherlandStock_ExRet']]
netherland1 = netherland1[~np.isnan(lag_Stock_Index)]
netherland1.head()
New Zealand
In [ ]:
newzealand = data.copy()
newzealand = newzealand[['newzealandltgovbondyield10y', 'newzealand3m', 'newzealandcpi', 'newzealanddivyld']]
newzealand.head()
#Monthly log dividend yield
newzealand['newzealandStock_DY'] = newzealand['newzealanddivyld']
newzealand['newzealandLog_Stock_DY'] = np.log(newzealand['newzealandStock_DY'])
#Monthly lagged stock index
newzealand['newzealandStock_Index'] = newzealand['newzealandcpi']
newzealand['newzealandlag_Stock_Index'] = (newzealand['newzealandStock_Index']).shift(1)
lag_Stock_Index = newzealand['newzealandlag_Stock_Index']
#Stock Return
newzealand['newzealandStock_Ret'] = np.log(newzealand['newzealandStock_Index']/newzealand['newzealandlag_Stock_Index']+newzealand['newzealandStock_DY'])
#Stock Excess Return
newzealand['newzealandStock_ExRet'] = newzealand['newzealandStock_Ret'] - norway['RiskFree']
newzealand1 = newzealand[['newzealandLog_Stock_DY','newzealandlag_Stock_Index','newzealandStock_Ret','newzealandStock_ExRet']]
newzealand1 = newzealand1[~np.isnan(lag_Stock_Index)]
newzealand1.head()
Poland
In [ ]:
poland = data.copy()
poland = poland[['polandltgovbondyield10y', 'poland3m', 'polandcpi', 'polanddivyld']]
poland.head()
#Monthly log dividend yield
poland['polandStock_DY'] = poland['polanddivyld']
poland['polandLog_Stock_DY'] = np.log(poland['polandStock_DY'])
#Monthly lagged stock index
poland['polandStock_Index'] = poland['polandcpi']
poland['polandlag_Stock_Index'] = (poland['polandStock_Index']).shift(1)
lag_Stock_Index = poland['polandlag_Stock_Index']
#Stock Return
```

```
        poland['polandStock_Ret'] = np.log(poland['polandStock_Index']/poland['polandlag_Stock_Index']+poland['polandStock_DY'])

        #Stock Excess Return
        poland['polandStock_ExRet'] = poland['polandStock_Ret'] - norway['RiskFree']

        poland1 = poland[['polandLog_Stock_DY','polandlag_Stock_Index','polandStock_Ret','polandStock_ExRet']]
        poland1 = poland1[~np.isnan(lag_Stock_Index)]
        poland1.head()
        Portugal
        In [ ]:

        portugal = data.copy()
        portugal = portugal[['portugalltgovbondyield10y', 'portugal3m', 'portugalcpi', 'portugaldivyld']]
        portugal.head()

        #Monthly log dividend yield
        portugal['portugalStock_DY'] = portugal['portugaldivyld']
        portugal['portugalLog_Stock_DY'] = np.log(portugal['portugalStock_DY'])

        #Monthly lagged stock index
        portugal['portugalStock_Index'] = portugal['portugalcpi']
        portugal['portugallag_Stock_Index'] = (portugal['portugalStock_Index']).shift(1)
        lag_Stock_Index = portugal['portugallag_Stock_Index']

        #Stock Return
        portugal['portugalStock_Ret'] = np.log(portugal['portugalStock_Index']/portugal['portugallag_Stock_Index']+portugal['portugalStock_DY'])

        #Stock Excess Return
        portugal['portugalStock_ExRet'] = portugal['portugalStock_Ret'] - norway['RiskFree']

        portugal1 = portugal[['portugalLog_Stock_DY','portugallag_Stock_Index','portugalStock_Ret','portugalStock_ExRet']]
        portugal1 = portugal1[~np.isnan(lag_Stock_Index)]
        portugal1.head()
        Russia
        In [ ]:

        portugal = data.copy()
        portugal = portugal[['portugalltgovbondyield10y', 'portugal3m', 'portugalcpi', 'portugaldivyld']]
        portugal.head()

        #Monthly log dividend yield
        portugal['portugalStock_DY'] = portugal['portugaldivyld']
        portugal['portugalLog_Stock_DY'] = np.log(portugal['portugalStock_DY'])

        #Monthly lagged stock index
        portugal['portugalStock_Index'] = portugal['portugalcpi']
        portugal['portugallag_Stock_Index'] = (portugal['portugalStock_Index']).shift(1)
        lag_Stock_Index = portugal['portugallag_Stock_Index']

        #Stock Return
        portugal['portugalStock_Ret'] = np.log(portugal['portugalStock_Index']/portugal['portugallag_Stock_Index']+portugal['portugalStock_DY'])
```

```

#Stock Excess Return
portugal['portugalStock_ExRet'] = portugal['portugalStock_Ret'] - norway['RiskFree']

portugal1 = portugal[['portugalLog_Stock_DY','portugallag_Stock_Index','portugalStock_Ret','portugalStock_ExRet']]
portugal1 = portugal1[~np.isnan(lag_Stock_Index)]
portugal1.head()
South Africa
In [ ]:

southafrica = data.copy()
southafrica = southafrica[['southafricaltgovbondyield10y', 'southafrica3m', 'southafricacpi', 'southafricadivyld']]
southafrica.head()

#Monthly log dividend yield
southafrica['southafricaStock_DY'] = southafrica['southafricadivyld']
southafrica['southafricaLog_Stock_DY'] = np.log(southafrica['southafricaStock_DY'])

#Monthly lagged stock index
southafrica['southafricaStock_Index'] = southafrica['southafricacpi']
southafrica['southafricalag_Stock_Index'] = (southafrica['southafricaStock_Index']).shift(1)
lag_Stock_Index = southafrica['southafricalag_Stock_Index']

#Stock Return
southafrica['southafricaStock_Ret'] = np.log(southafrica['southafricaStock_Index']/southafrica['southafricalag_Stock_Index']+southafrica['southafricaStock_DY'])

#Stock Excess Return
southafrica['southafricaStock_ExRet'] = southafrica['southafricaStock_Ret'] - norway['RiskFree']

southafrica1 = southafrica[['southafricaLog_Stock_DY','southafricalag_Stock_Index','southafricaStock_Ret','southafricaStock_ExRet']]
southafrica1 = southafrica1[~np.isnan(lag_Stock_Index)]
southafrica1.head()
South Korea
In [ ]:

southkorea = data.copy()
southkorea = southkorea[['southkorealtgovbondyield10y', 'southkorea3m', 'southkoreacpi', 'southkoreadivyld']]
southkorea.head()

#Monthly log dividend yield
southkorea['southkoreaStock_DY'] = southkorea['southkoreadivyld']
southkorea['southkoreaLog_Stock_DY'] = np.log(southkorea['southkoreaStock_DY'])

#Monthly lagged stock index
southkorea['southkoreaStock_Index'] = southkorea['southkoreacpi']
southkorea['southkorealag_Stock_Index'] = (southkorea['southkoreaStock_Index']).shift(1)
lag_Stock_Index = southkorea['southkorealag_Stock_Index']

#Stock Return
```

```
southkorea['southkoreaStock_Ret'] = np.log(southkorea['southkoreaStock_Index']/southkorea['southkorealag_Stock_Index']+southkorea['southkoreaStock_DY'])

#Stock Excess Return
southkorea['southkoreaStock_ExRet'] = southkorea['southkoreaStock_Ret'] - norway['RiskFree']

southkorea1 = southkorea[['southkoreaLog_Stock_DY','southkorealag_Stock_Index','southkoreaStock_Ret','southkoreaStock_ExRet']]
southkorea1 = southkorea1[~np.isnan(lag_Stock_Index)]
southkorea1.head()
Spain
In [ ]:

spain = data.copy()
spain = spain[['spainltgovbondyield10y', 'spain3m', 'spaincpi', 'spaindivyld']]
spain.head()

#Monthly log dividend yield
spain['spainStock_DY'] = spain['spaindivyld']
spain['spainLog_Stock_DY'] = np.log(spain['spainStock_DY'])

#Monthly lagged stock index
spain['spainStock_Index'] = spain['spaincpi']
spain['spainlag_Stock_Index'] = (spain['spainStock_Index']).shift(1)
lag_Stock_Index = spain['spainlag_Stock_Index']

#Stock Return
spain['spainStock_Ret'] = np.log(spain['spainStock_Index']/spain['spainlag_Stock_Index']+spain['spainStock_DY'])

#Stock Excess Return
spain['spainStock_ExRet'] = spain['spainStock_Ret'] - norway['RiskFree']

spain1 = spain[['spainLog_Stock_DY','spainlag_Stock_Index','spainStock_Ret','spainStock_ExRet']]
spain1 = spain1[~np.isnan(lag_Stock_Index)]
spain1.head()
Sweden
In [ ]:

sweden = data.copy()
sweden = sweden[['swedenltgovbondyield10y', 'sweden3m', 'swedencpi', 'swedendivyld']]
sweden.head()

#Monthly log dividend yield
sweden['swedenStock_DY'] = sweden['swedendivyld']
sweden['swedenLog_Stock_DY'] = np.log(sweden['swedenStock_DY'])

#Monthly lagged stock index
sweden['swedenStock_Index'] = sweden['swedencpi']
sweden['swedenlag_Stock_Index'] = (sweden['swedenStock_Index']).shift(1)
lag_Stock_Index = sweden['swedenlag_Stock_Index']

#Stock Return
sweden['swedenStock_Ret'] = np.log(sweden['swedenStock_Index']/sweden['swedenlag_Stock_Index']+sweden['swedenStock_DY'])

#Stock Excess Return
```

```
sweden['swedenStock_ExRet'] = sweden['swedenStock_Ret'] - norway['RiskFree']

sweden1 = sweden[['swedenLog_Stock_DY','swedenlag_Stock_Index','swedenStock_Ret','swedenStock_ExRet']]
sweden1 = sweden1[~np.isnan(lag_Stock_Index)]
sweden1.head()
Switzerland
In [ ]:

switzerland = data.copy()
switzerland = switzerland[['switzerlandltgovbondyield10y', 'switzerland3m', 'switzerlandcpi', 'switzerlanddivyld']]
switzerland.head()

#Monthly log dividend yield
switzerland['switzerlandStock_DY'] = switzerland['switzerlanddivyld']
switzerland['switzerlandLog_Stock_DY'] = np.log(switzerland['switzerlandStock_DY'])

#Monthly lagged stock index
switzerland['switzerlandStock_Index'] = switzerland['switzerlandcpi']
switzerland['switzerlandlag_Stock_Index'] = (switzerland['switzerlandStock_Index']).shift(1)
lag_Stock_Index = switzerland['switzerlandlag_Stock_Index']

#Stock Return
switzerland['switzerlandStock_Ret'] = np.log(switzerland['switzerlandStock_Index']/switzerland['switzerlandlag_Stock_Index']+switzerland['switzerlandStock_DY'])

#Stock Excess Return
switzerland['switzerlandStock_ExRet'] = switzerland['switzerlandStock_Ret'] - norway['RiskFree']

switzerland1 = switzerland[['switzerlandLog_Stock_DY','switzerlandlag_Stock_Index','switzerlandStock_Ret','switzerlandStock_ExRet']]
switzerland1 = switzerland1[~np.isnan(lag_Stock_Index)]
switzerland1.head()
United Kingdom
In [ ]:

uk = data.copy()
uk = uk[['ukltgovbondyield10y', 'uk3m', 'ukcpi', 'ukdivyld']]
uk.head()

#Monthly log dividend yield
uk['ukStock_DY'] = uk['ukdivyld']
uk['ukLog_Stock_DY'] = np.log(uk['ukStock_DY'])

#Monthly lagged stock index
uk['ukStock_Index'] = uk['ukcpi']
uk['uklag_Stock_Index'] = (uk['ukStock_Index']).shift(1)
lag_Stock_Index = uk['uklag_Stock_Index']

#Stock Return
uk['ukStock_Ret'] = np.log(uk['ukStock_Index']/uk['uklag_Stock_Index']+uk['ukStock_DY'])

#Stock Excess Return
uk['ukStock_ExRet'] = uk['ukStock_Ret'] - norway['RiskFree']

uk1 = uk[['ukLog_Stock_DY','uklag_Stock_Index','ukStock_Ret','ukStock_ExRet']]
uk1 = uk1[~np.isnan(lag_Stock_Index)]
```

```
uk1.head()
United states
In [ ]:

us = data.copy()
us = us[['usltgovbondyield10y', 'us3m', 'uscpi', 'usdivyld']]

#Monthly log dividend yield
us['usStock_DY'] = us['usdivyld']
us['usLog_Stock_DY'] = np.log(us['usStock_DY'])

#Monthly lagged stock index
us['usStock_Index'] = us['uscpi']
us['uslag_Stock_Index'] = (us['usStock_Index']).shift(1)
lag_Stock_Index = us['uslag_Stock_Index']

#Stock Return
us['usStock_Ret'] = np.log(us['usStock_Index']/us['uslag_Stock_Index']+us['usStock_DY'])

#Stock Excess Return
us['usStock_ExRet'] = us['usStock_Ret'] - norway['RiskFree']

us1 = us[['usLog_Stock_DY','uslag_Stock_Index','usStock_Ret','usStock_ExRet']]
us1 = us1[~np.isnan(lag_Stock_Index)]
us1.head()
Constant Expected Return (CER) Analysis
In [222]:

df = pd.concat([austria.loc['2008-08-01':'2018-04-01'],
                belgium1.loc['2008-08-01':'2018-04-01'],
                canada.loc['2008-08-01':'2018-04-01'],
                chechrepublic.loc['2008-08-01':'2018-04-01'],
                chile.loc['2008-08-01':'2018-04-01'],
                china.loc['2008-08-01':'2018-04-01'],
                denmark.loc['2008-08-01':'2018-04-01'],
                france.loc['2008-08-01':'2018-04-01'],
                germany.loc['2008-08-01':'2018-04-01'],
                hungary.loc['2008-08-01':'2018-04-01'],
                israel.loc['2008-08-01':'2018-04-01'],
                italy.loc['2008-08-01':'2018-04-01'],
                japan.loc['2008-08-01':'2018-04-01'],
                mexico.loc['2008-08-01':'2018-04-01'],
                netherland.loc['2008-08-01':'2018-04-01'],
                poland.loc['2008-08-01':'2018-04-01'],
                portugal.loc['2008-08-01':'2018-04-01'],
                russia.loc['2008-08-01':'2018-04-01'],
                southafrica.loc['2008-08-01':'2018-04-01'],
                southkorea.loc['2008-08-01':'2018-04-01'],
                spain.loc['2008-08-01':'2018-04-01'],
                sweden.loc['2008-08-01':'2018-04-01'],
                switzerland.loc['2008-08-01':'2018-04-01'],
                uk.loc['2008-08-01':'2018-04-01'],
                us.loc['2008-08-01':'2018-04-01']], axis=1)
In [ ]:

CERportfolio = df[['austriaStock_ExRet',
        'belgiumStock_ExRet',
        'canadaStock_ExRet',
        'chechrepublicStock_ExRet',
        'chileStock_ExRet',
        'chinaStock_ExRet',
        'denmarkStock_ExRet',
        'franceStock_ExRet',
        'germanyStock_ExRet',
```

```
                'hungaryStock_ExRet',
                'israelStock_ExRet',
                'italyStock_ExRet',
                'japanStock_ExRet',
                'mexicoStock_ExRet',
                'netherlandStock_ExRet',
                'polandStock_ExRet',
                'portugalStock_ExRet',
                'russiaStock_ExRet',
                'southafricaStock_ExRet',
                'southkoreaStock_ExRet',
                'spainStock_ExRet',
                'swedenStock_ExRet',
                'switzerlandStock_ExRet',
                'ukStock_ExRet',
                'usStock_ExRet']]

        plt.figure(figsize=(20, 10))
        plt.plot(CERportfolio, alpha=20);
        plt.xlabel('Date')
        plt.ylabel('Returns')
        In [ ]:

        #Compute the cumulative sum of excess returns
        Ri = np.log(CERportfolio).diff().diff().dropna()*12
        Ri = pd.DataFrame(Ri)

        plt.figure(figsize=(20, 10))
        plt.plot(Ri, alpha=20);
        plt.xlabel('Date')
        plt.ylabel('Returns')
        A common way to present the "effect" of a mutual fund's performan
    ce over time is to show the cumulative return with a visual such as a mo
    untain graph.
        In [ ]:

        CERportfolio.kurtosis()
        In [ ]:

        CERportfolio.skew()
        In [227]:

        norwayRiskFree = norwayRiskFree.loc['2008-08-01':'2018-04-01']
        In [228]:

        Rreturns_annual = CERportfolio.mean()
        cov_annuals = CERportfolio*100
        Rcov_annual = cov_annuals.cov()
        In [229]:

        # empty lists to store returns, volatility and weights of imigina
    ry portfolios
        Rport_returns = []
        Rport_volatility = []
        Rsharpe_ratio = []
        Rstock_weights = []
        In [230]:

        # set the number of combinations for imaginary portfolios
        Rnum_assets = 25
        Rnum_portfolios = 50000
        In [231]:

        #set random seed for reproduction's sake
        np.random.seed(101)
        In [232]:

```

```
        # populate the empty lists with each portfolios returns,risk and weights
        for single_portfolio in range(Rnum_portfolios):
            weights = np.random.random(Rnum_assets)
            weights /= np.sum(weights)
            returns = np.dot(weights, Rreturns_annual)
            volatility = np.sqrt(np.dot(weights.T, np.dot(Rcov_annual, weights)))
            sharpe = returns/ volatility
            Rsharpe_ratio.append(sharpe)
            Rport_returns.append(returns)
            Rport_volatility.append(volatility)
            Rstock_weights.append(weights)
        In [ ]:

        sum(weights)
        In [234]:

        # a dictionary for Returns and Risk values of each portfolio
        portfolio = {'Returns': Rport_returns,
                     'Volatility': Rport_volatility,
                     'Sharpe Ratio': Rsharpe_ratio}
        In [235]:

        # extend original dictionary to accomodate each ticker and weight in the portfolio
        for counter,symbol in enumerate(CERportfolio):
            portfolio[symbol+' Weight'] = [Weight[counter] for Weight in Rstock_weights]
        In [236]:

        # make a nice dataframe of the extended dictionary
        df1 = pd.DataFrame(portfolio)
        In [237]:

        # get better labels for desired arrangement of columns
        column_order = ['Returns', 'Volatility','Sharpe Ratio'] + [stock+' Weight' for stock in CERportfolio ]
        In [ ]:

        # reorder dataframe columns
        df1 = df1[column_order]
        df1.head()
        In [ ]:

        # plot frontier, max sharpe & min Volatility values with a scatterplot
        plt.style.use('seaborn-dark')
        df1.plot.scatter(x='Volatility', y='Returns', c='Sharpe Ratio', cmap='RdYlGn',
                                edgecolors='black', figsize=(10, 10), grid=True)

        plt.xlabel('Volatility(Std. Deviation)')
        plt.ylabel('Expected Returns')
        plt.title('Efficient Frontier')
        plt.show()
        In [240]:

        # find min Volatility & max sharpe values in the dataframe (df)
        min_volatility1 = df1['Volatility'].min()
        max_sharpe1 = df1['Sharpe Ratio'].max()
        In [241]:

        # use the min, max values to locate and create the two special portfolios
        sharpe_portfolio1 = df1.loc[df1['Sharpe Ratio'] == max_sharpe1]
```

```
min_variance_port1 = df1.loc[df1['Volatility'] == min_volatility1]
In [ ]:

## plot frontier, max sharpe & min Volatility values with a scatterplot
plt.style.use('seaborn-dark')
df1.plot.scatter(x='Volatility', y='Returns', c='Sharpe Ratio',
                cmap='RdYlGn', edgecolors='black', figsize=(10, 10), grid=True)
plt.scatter(x=sharpe_portfolio1['Volatility'], y=sharpe_portfolio1['Returns'], c='red', marker='D', s=200)
plt.scatter(x=min_variance_port1['Volatility'], y=min_variance_port1['Returns'], c='blue', marker='D', s=200 )
plt.ylabel('Expected Returns')
plt.xlabel('Volatility (Std. Deviation)')
plt.title('Efficient Frontier')
plt.show()
In [ ]:

# print the details of the 2 special portfolios
print(min_variance_port1.T)
In [ ]:

print(sharpe_portfolio1.T)
Time Varying Expected Return (TVER) Analysis
Create returns at different frequencies
In [245]:

austriaStock_Ret1y = austria['austriaStock_Ret'].rolling(window=12).mean()*12
austriaStock_Ret3y = austria['austriaStock_Ret'].rolling(window=36).mean()*36
austriaStock_Ret3Ya = austriaStock_Ret3y/3

belgiumStock_Ret1y = belgium['belgiumStock_Ret'].rolling(window=12).mean()*12
belgiumStock_Ret3y = belgium['belgiumStock_Ret'].rolling(window=36).mean()*36
belgiumStock_Ret3Ya = belgiumStock_Ret3y/3

canadaStock_Ret1y = canada['canadaStock_Ret'].rolling(window=12).mean()*12
canadaStock_Ret3y = canada['canadaStock_Ret'].rolling(window=36).mean()*36
canadaStock_Ret3Ya = canadaStock_Ret3y/3

chechrepublicStock_Ret1y = chechrepublic['chechrepublicStock_Ret'].rolling(window=12).mean()*12
chechrepublicStock_Ret3y = chechrepublic['chechrepublicStock_Ret'].rolling(window=36).mean()*36
chechrepublicStock_Ret3Ya = chechrepublicStock_Ret3y/3

chileStock_Ret1y = chile['chileStock_Ret'].rolling(window=12).mean()*12
chileStock_Ret3y = chile['chileStock_Ret'].rolling(window=36).mean()*36
chileStock_Ret3Ya = chileStock_Ret3y/3

chinaStock_Ret1y = china['chinaStock_Ret'].rolling(window=12).mean()*12
chinaStock_Ret3y = china['chinaStock_Ret'].rolling(window=36).mean()*36
chinaStock_Ret3Ya = chinaStock_Ret3y/3

denmarkStock_Ret1y = denmark['denmarkStock_Ret'].rolling(window=12).mean()*12
```

```
        denmarkStock_Ret3y = denmark['denmarkStock_Ret'].rolling(window=36).mean()*36
        denmarkStock_Ret3Ya = denmarkStock_Ret3y/3

        franceStock_Ret1y = france['franceStock_Ret'].rolling(window=12).mean()*12
        franceStock_Ret3y = france['franceStock_Ret'].rolling(window=36).mean()*36
        franceStock_Ret3Ya = franceStock_Ret3y/3

        germanyStock_Ret1y = germany['germanyStock_Ret'].rolling(window=12).mean()*12
        germanyStock_Ret3y = germany['germanyStock_Ret'].rolling(window=36).mean()*36
        germanyStock_Ret3Ya = germanyStock_Ret3y/3

        hungaryStock_Ret1y = hungary['hungaryStock_Ret'].rolling(window=12).mean()*12
        hungaryStock_Ret3y = hungary['hungaryStock_Ret'].rolling(window=36).mean()*36
        hungaryStock_Ret3Ya = hungaryStock_Ret3y/3

        israelStock_Ret1y = israel['israelStock_Ret'].rolling(window=12).mean()*12
        israelStock_Ret3y = israel['israelStock_Ret'].rolling(window=36).mean()*36
        israelStock_Ret3Ya = israelStock_Ret3y/3

        italyStock_Ret1y = italy['italyStock_Ret'].rolling(window=12).mean()*12
        italyStock_Ret3y = italy['italyStock_Ret'].rolling(window=36).mean()*36
        italyStock_Ret3Ya = italyStock_Ret3y/3

        japanStock_Ret1y = japan['japanStock_Ret'].rolling(window=12).mean()*12
        japanStock_Ret3y = japan['japanStock_Ret'].rolling(window=36).mean()*36
        japanStock_Ret3Ya = japanStock_Ret3y/3

        mexicoStock_Ret1y = mexico['mexicoStock_Ret'].rolling(window=12).mean()*12
        mexicoStock_Ret3y = mexico['mexicoStock_Ret'].rolling(window=36).mean()*36
        mexicoStock_Ret3Ya = mexicoStock_Ret3y/3

        netherlandStock_Ret1y = netherland['netherlandStock_Ret'].rolling(window=12).mean()*12
        netherlandStock_Ret3y = netherland['netherlandStock_Ret'].rolling(window=36).mean()*36
        netherlandStock_Ret3Ya = netherlandStock_Ret3y/3

        polandStock_Ret1y = poland['polandStock_Ret'].rolling(window=12).mean()*12
        polandStock_Ret3y = poland['polandStock_Ret'].rolling(window=36).mean()*36
        polandStock_Ret3Ya = polandStock_Ret3y/3

        portugalStock_Ret1y = portugal['portugalStock_Ret'].rolling(window=12).mean()*12
        portugalStock_Ret3y = portugal['portugalStock_Ret'].rolling(window=36).mean()*36
        portugalStock_Ret3Ya = portugalStock_Ret3y/3

        russiaStock_Ret1y = russia['russiaStock_Ret'].rolling(window=12).mean()*12
```

```
        russiaStock_Ret3y = russia['russiaStock_Ret'].rolling(window=36).mean()*36
        russiaStock_Ret3Ya = russiaStock_Ret3y/3

        southafricaStock_Ret1y = southafrica['southafricaStock_Ret'].rolling(window=12).mean()*12
        southafricaStock_Ret3y = southafrica['southafricaStock_Ret'].rolling(window=36).mean()*36
        southafricaStock_Ret3Ya = southafricaStock_Ret3y/3

        southkoreaStock_Ret1y = southkorea['southkoreaStock_Ret'].rolling(window=12).mean()*12
        southkoreaStock_Ret3y = southkorea['southkoreaStock_Ret'].rolling(window=36).mean()*36
        southkoreaStock_Ret3Ya = southkoreaStock_Ret3y/3

        spainStock_Ret1y = spain['spainStock_Ret'].rolling(window=12).mean()*12
        spainStock_Ret3y = spain['spainStock_Ret'].rolling(window=36).mean()*36
        spainStock_Ret3Ya = spainStock_Ret3y/3

        swedenStock_Ret1y = sweden['swedenStock_Ret'].rolling(window=12).mean()*12
        swedenStock_Ret3y = sweden['swedenStock_Ret'].rolling(window=36).mean()*36
        swedenStock_Ret3Ya = swedenStock_Ret3y/3

        switzerlandStock_Ret1y = switzerland['switzerlandStock_Ret'].rolling(window=12).mean()*12
        switzerlandStock_Ret3y = switzerland['switzerlandStock_Ret'].rolling(window=36).mean()*36
        switzerlandStock_Ret3Ya = switzerlandStock_Ret3y/3

        ukStock_Ret1y = uk['ukStock_Ret'].rolling(window=12).mean()*12
        ukStock_Ret3y = uk['ukStock_Ret'].rolling(window=36).mean()*36
        ukStock_Ret3Ya = ukStock_Ret3y/3

        usStock_Ret1y = us['usStock_Ret'].rolling(window=12).mean()*12
        usStock_Ret3y = us['usStock_Ret'].rolling(window=36).mean()*36
        usStock_Ret3Ya = usStock_Ret3y/3

        norway_RF_Ret3y = norwayRiskFree.rolling(window=36).mean()*36
        Excess returns 3Y horizon
        In [246]:

        austriaStock_ExRet3Y = austriaStock_Ret3y - norway_RF_Ret3y
        belgiumStock_ExRet3Y = belgiumStock_Ret3y - norway_RF_Ret3y
        canadaStock_ExRet3Y = canadaStock_Ret3y - norway_RF_Ret3y
        chechrepublicStock_ExRet3Y = chechrepublicStock_Ret3y - norway_RF_Ret3y
        chileStock_ExRet3Y = chileStock_Ret3y - norway_RF_Ret3y
        chinaStock_ExRet3Y = chinaStock_Ret3y - norway_RF_Ret3y
        denmarkStock_ExRet3Y = denmarkStock_Ret3y - norway_RF_Ret3y
        franceStock_ExRet3Y = franceStock_Ret3y - norway_RF_Ret3y
        germanyStock_ExRet3Y = germanyStock_Ret3y - norway_RF_Ret3y
        hungaryStock_ExRet3Y = hungaryStock_Ret3y - norway_RF_Ret3y
        isrealStock_ExRet3Y = israelStock_Ret3y - norway_RF_Ret3y
        italyStock_ExRet3Y = italyStock_Ret3y - norway_RF_Ret3y
        japanStock_ExRet3Y = japanStock_Ret3y - norway_RF_Ret3y
        mexicoStock_ExRet3Y = mexicoStock_Ret3y - norway_RF_Ret3y
        netherlandStock_ExRet3Y = netherlandStock_Ret3y - norway_RF_Ret3y
        polandStock_ExRet3Y = polandStock_Ret3y - norway_RF_Ret3y
        portugalStock_ExRet3Y = portugalStock_Ret3y - norway_RF_Ret3y
        russiaStock_ExRet3Y = russiaStock_Ret3y - norway_RF_Ret3y
```

```
southafricaStock_ExRet3Y = southafricaStock_Ret3y - norway_RF_Ret3y
southkoreaStock_ExRet3Y = southkoreaStock_Ret3y - norway_RF_Ret3y
spainStock_ExRet3Y = spainStock_Ret3y - norway_RF_Ret3y
swedenStock_ExRet3Y = swedenStock_Ret3y - norway_RF_Ret3y
switzerlandStock_ExRet3Y = switzerlandStock_Ret3y - norway_RF_Ret3y
ukStock_ExRet3Y = ukStock_Ret3y - norway_RF_Ret3y
usStock_ExRet3Y = usStock_Ret3y - norway_RF_Ret3y
Make dataframes of the Excess returns 3Y horizon
In [247]:
austriaStock_ExRet3Y = pd.DataFrame(austriaStock_ExRet3Y)
belgiumStock_ExRet3Y = pd.DataFrame(belgiumStock_ExRet3Y)
canadaStock_ExRet3Y = pd.DataFrame(canadaStock_ExRet3Y)
chechrepublicStock_ExRet3Y = pd.DataFrame(chechrepublicStock_ExRet3Y)
chechrepublicStock_ExRet3Y = pd.DataFrame(chechrepublicStock_ExRet3Y)
chileStock_ExRet3Y = pd.DataFrame(chileStock_ExRet3Y)
chinaStock_ExRet3Y = pd.DataFrame(chinaStock_ExRet3Y)
denmarkStock_ExRet3Y = pd.DataFrame(denmarkStock_ExRet3Y)
franceStock_ExRet3Y = pd.DataFrame(franceStock_ExRet3Y)
germanyStock_ExRet3Y = pd.DataFrame(germanyStock_ExRet3Y)
hungaryStock_ExRet3Y = pd.DataFrame(hungaryStock_ExRet3Y)
isrealStock_ExRet3Y = pd.DataFrame(isrealStock_ExRet3Y)
italyStock_ExRet3Y = pd.DataFrame(italyStock_ExRet3Y)
japanStock_ExRet3Y = pd.DataFrame(japanStock_ExRet3Y)
mexicoStock_ExRet3Y = pd.DataFrame(mexicoStock_ExRet3Y)
netherlandStock_ExRet3Y = pd.DataFrame(netherlandStock_ExRet3Y)
polandStock_ExRet3Y = pd.DataFrame(polandStock_ExRet3Y)
portugalStock_ExRet3Y = pd.DataFrame(portugalStock_ExRet3Y)
russiaStock_ExRet3Y = pd.DataFrame(russiaStock_ExRet3Y)
southafricaStock_ExRet3Y = pd.DataFrame(southafricaStock_ExRet3Y)
southkoreaStock_ExRet3Y = pd.DataFrame(southkoreaStock_ExRet3Y)
spainStock_ExRet3Y = pd.DataFrame(spainStock_ExRet3Y)
swedenStock_ExRet3Y = pd.DataFrame(swedenStock_ExRet3Y)
switzerlandStock_ExRet3Y = pd.DataFrame(switzerlandStock_ExRet3Y)
ukStock_ExRet3Y = pd.DataFrame(ukStock_ExRet3Y)
usStock_ExRet3Y = pd.DataFrame(usStock_ExRet3Y)
In [248]:
austriaStock_ExRet3Y.columns = ['austriaStock_ExRet3Y']
belgiumStock_ExRet3Y.columns = ['belgiumStock_ExRet3Y']
canadaStock_ExRet3Y.columns = ['canadaStock_ExRet3Y']
chechrepublicStock_ExRet3Y.columns = ['checkrepublicStock_ExRet3Y']
chileStock_ExRet3Y.columns = ['chileStock_ExRet3Y']
chinaStock_ExRet3Y.columns = ['chinaStock_ExRet3Y']
denmarkStock_ExRet3Y.columns = ['denmarkStock_ExRet3Y']
franceStock_ExRet3Y.columns = ['franceStock_ExRet3Y']
germanyStock_ExRet3Y.columns = ['germanyStock_ExRet3Y']
hungaryStock_ExRet3Y.columns = ['hungaryStock_ExRet3Y']
isrealStock_ExRet3Y.columns = ['isrealStock_ExRet3Y']
italyStock_ExRet3Y.columns = ['italyStock_ExRet3Y']
japanStock_ExRet3Y.columns = ['japanStock_ExRet3Y']
mexicoStock_ExRet3Y.columns = ['mexicoStock_ExRet3Y']
netherlandStock_ExRet3Y.columns = ['netherlandStock_ExRet3Y']
polandStock_ExRet3Y.columns = ['polandStock_ExRet3Y']
portugalStock_ExRet3Y.columns = ['portugalStock_ExRet3Y']
russiaStock_ExRet3Y.columns = ['russiaStock_ExRet3Y']
southafricaStock_ExRet3Y.columns = ['aouthafricaStock_ExRet3Y']
southkoreaStock_ExRet3Y.columns = ['southkoreaStock_ExRet3Y']
spainStock_ExRet3Y.columns = ['spainStock_ExRet3Y']
```

```
swedenStock_ExRet3Y.columns = ['swedenStock_ExRet3Y']
switzerlandStock_ExRet3Y.columns = ['switzerlandStock_ExRet3Y']
ukStock_ExRet3Y.columns = ['ukStock_ExRet3Y']
usStock_ExRet3Y.columns = ['usStock_ExRet3Y']
Return decomposition 3Y horizon
In [249]:

austriaStock_CG3Y = (np.log((austria['austriaStock_Index'])/(austria['austriaStock_Index']).shift())).rolling(window=36).mean()*36
belgiumStock_CG3Y = (np.log((belgium['belgiumStock_Index'])/(belgium['belgiumStock_Index']).shift())).rolling(window=36).mean()*36
canadaStock_CG3Y = (np.log((canada['canadaStock_Index'])/(canada['canadaStock_Index']).shift())).rolling(window=36).mean()*36
chechrepublicStock_CG3Y = (np.log((chechrepublic['chechrepublicStock_Index'])/(chechrepublic['chechrepublicStock_Index']).shift())).rolling(window=36).mean()*36
chileStock_CG3Y = (np.log((chile['chileStock_Index'])/(chile['chileStock_Index']).shift())).rolling(window=36).mean()*36
chinaStock_CG3Y = (np.log((china['chinaStock_Index'])/(china['chinaStock_Index']).shift())).rolling(window=36).mean()*36
denmarkStock_CG3Y = (np.log((denmark['denmarkStock_Index'])/(denmark['denmarkStock_Index']).shift())).rolling(window=36).mean()*36
franceStock_CG3Y = (np.log((france['franceStock_Index'])/(france['franceStock_Index']).shift())).rolling(window=36).mean()*36
germanyStock_CG3Y = (np.log((germany['germanyStock_Index'])/(germany['germanyStock_Index']).shift())).rolling(window=36).mean()*36
hungaryStock_CG3Y = (np.log((hungary['hungaryStock_Index'])/(hungary['hungaryStock_Index']).shift())).rolling(window=36).mean()*36
israelStock_CG3Y = (np.log((israel['israelStock_Index'])/(israel['israelStock_Index']).shift())).rolling(window=36).mean()*36
italyStock_CG3Y = (np.log((italy['italyStock_Index'])/(italy['italyStock_Index']).shift())).rolling(window=36).mean()*36
japanStock_CG3Y = (np.log((japan['japanStock_Index'])/(japan['japanStock_Index']).shift())).rolling(window=36).mean()*36
mexicoStock_CG3Y = (np.log((mexico['mexicoStock_Index'])/(mexico['mexicoStock_Index']).shift())).rolling(window=36).mean()*36
netherlandStock_CG3Y = (np.log((netherland['netherlandStock_Index'])/(netherland['netherlandStock_Index']).shift())).rolling(window=36).mean()*36
polandStock_CG3Y = (np.log((poland['polandStock_Index'])/(poland['polandStock_Index']).shift())).rolling(window=36).mean()*36
portugalStock_CG3Y = (np.log((portugal['portugalStock_Index'])/(portugal['portugalStock_Index']).shift())).rolling(window=36).mean()*36

russiaStock_CG3Y = (np.log((russia['russiaStock_Index'])/(russia['russiaStock_Index']).shift())).rolling(window=36).mean()*36
southafricaStock_CG3Y = (np.log((southafrica['southafricaStock_Index'])/(southafrica['southafricaStock_Index']).shift())).rolling(window=36).mean()*36
southkoreaStock_CG3Y = (np.log((southkorea['southkoreaStock_Index'])/(southkorea['southkoreaStock_Index']).shift())).rolling(window=36).mean()*36
spainStock_CG3Y = (np.log((spain['spainStock_Index'])/(spain['spainStock_Index']).shift())).rolling(window=36).mean()*36
swedenStock_CG3Y = (np.log((sweden['swedenStock_Index'])/(sweden['swedenStock_Index']).shift())).rolling(window=36).mean()*36
switzerlandStock_CG3Y = (np.log((switzerland['switzerlandStock_Index'])/(switzerland['switzerlandStock_Index']).shift())).rolling(window=36).mean()*36
ukStock_CG3Y = (np.log((uk['ukStock_Index'])/(uk['ukStock_Index']).shift())).rolling(window=36).mean()*36
usStock_CG3Y = (np.log((us['usStock_Index'])/(us['usStock_Index']).shift())).rolling(window=36).mean()*36
Portfolio Allocation with univariate TS models
In [250]:

```

```
    Ret3y = pd.concat([norway_RF_Ret3y.loc['2008-07-01':'2018-04-01'],
                       austriaStock_Ret3y.loc['2008-07-01':'2018-04-01'],
                       belgiumStock_Ret3y.loc['2008-07-01':'2018-04-01'],
                       canadaStock_Ret3y.loc['2008-07-01':'2018-04-01'],
                       chechrepublicStock_Ret3y.loc['2008-07-01':'2018-04-01'],
                       chileStock_Ret3y.loc['2008-07-01':'2018-04-01'],
                       chinaStock_Ret3y.loc['2008-07-01':'2018-04-01'],
                       denmarkStock_Ret3y.loc['2008-07-01':'2018-04-01'],
                       franceStock_Ret3y.loc['2008-07-01':'2018-04-01'],
                       germanyStock_Ret3y.loc['2008-07-01':'2018-04-01'],
                       hungaryStock_Ret3y.loc['2008-07-01':'2018-04-01'],
                       israelStock_Ret3y.loc['2008-07-01':'2018-04-01'],
                       italyStock_Ret3y.loc['2008-07-01':'2018-04-01'],
                       japanStock_Ret3y.loc['2008-07-01':'2018-04-01'],
                       mexicoStock_Ret3y.loc['2008-07-01':'2018-04-01'],
                       netherlandStock_Ret3y.loc['2008-07-01':'2018-04-01'],
                       polandStock_Ret3y.loc['2008-07-01':'2018-04-01'],
                       portugalStock_Ret3y.loc['2008-07-01':'2018-04-01'],
                       russiaStock_Ret3y.loc['2008-07-01':'2018-04-01'],
                       southafricaStock_Ret3y.loc['2008-07-01':'2018-04-01'],
                       southkoreaStock_Ret3y.loc['2008-07-01':'2018-04-01'],
                       spainStock_Ret3y.loc['2008-07-01':'2018-04-01'],
                       swedenStock_Ret3y.loc['2008-07-01':'2018-04-01'],
                       switzerlandStock_Ret3y.loc['2008-07-01':'2018-04-01'],
                       ukStock_Ret3y.loc['2008-07-01':'2018-04-01'],
                       usStock_Ret3y.loc['2008-07-01':'2018-04-01']], axis=1)
    In [251]:

    Ret = pd.concat([norwayRiskFree.loc['2008-07-01':'2018-04-01'],
                     austria['austriaStock_Ret'].loc['2008-07-01':'2018-04-01'],
                     belgium['belgiumStock_Ret'].loc['2008-07-01':'2018-04-01'],
                     canada['canadaStock_Ret'].loc['2008-07-01':'2018-04-01'],
                     chechrepublic['chechrepublicStock_Ret'].loc['2008-07-01':'2018-04-01'],
                     chile['chileStock_Ret'].loc['2008-07-01':'2018-04-01'],
                     china['chinaStock_Ret'].loc['2008-07-01':'2018-04-01'],
```

```
                    denmark['denmarkStock_Ret'].loc['2008-07-01':'2018-04-01'],
                    france['franceStock_Ret'].loc['2008-07-01':'2018-04-01'],
                    germany['germanyStock_Ret'].loc['2008-07-01':'2018-04-01'],
                    hungary['hungaryStock_Ret'].loc['2008-07-01':'2018-04-01'],
                    israel['israelStock_Ret'].loc['2008-07-01':'2018-04-01'],
                    italy['italyStock_Ret'].loc['2008-07-01':'2018-04-01'],
                    japan['japanStock_Ret'].loc['2008-07-01':'2018-04-01'],
                    mexico['mexicoStock_Ret'].loc['2008-07-01':'2018-04-01'],
                    netherland['netherlandStock_Ret'].loc['2008-07-01':'2018-04-01'],
                    poland['polandStock_Ret'].loc['2008-07-01':'2018-04-01'],
                    portugal['portugalStock_Ret'].loc['2008-07-01':'2018-04-01'],
                    russia['russiaStock_Ret'].loc['2008-07-01':'2018-04-01'],
                    southafrica['southafricaStock_Ret'].loc['2008-07-01':'2018-04-01'],
                    southkorea['southkoreaStock_Ret'].loc['2008-07-01':'2018-04-01'],
                    spain['spainStock_Ret'].loc['2008-07-01':'2018-04-01'],
                    sweden['swedenStock_Ret'].loc['2008-07-01':'2018-04-01'],
                    switzerland['switzerlandStock_Ret'].loc['2008-07-01':'2018-04-01'],
                    uk['ukStock_Ret'].loc['2008-07-01':'2018-04-01'],
                    us['usStock_Ret'].loc['2008-07-01':'2018-04-01']],axis=1)
    In [252]:

    ExRet = pd.concat([austria['austriaStock_ExRet'].loc['2008-07-01':'2018-04-01'],
                     belgium['belgiumStock_ExRet'].loc['2008-07-01':'2018-04-01'],
                     canada['canadaStock_ExRet'].loc['2008-07-01':'2018-04-01'],
                     chechrepublic['chechrepublicStock_ExRet'].loc['2008-07-01':'2018-04-01'],
                     chile['chileStock_ExRet'].loc['2008-07-01':'2018-04-01'],
                     china['chinaStock_ExRet'].loc['2008-07-01':'2018-04-01'],
                     denmark['denmarkStock_ExRet'].loc['2008-07-01':'2018-04-01'],
                     france['franceStock_ExRet'].loc['2008-07-01':'2018-04-01'],
                     germany['germanyStock_ExRet'].loc['2008-07-01':'2018-04-01'],
                     hungary['hungaryStock_ExRet'].loc['2008-07-01':'2018-04-01'],
                     israel['israelStock_ExRet'].loc['2008-07-01':'2018-04-01'],
                     italy['italyStock_ExRet'].loc['2008-07-01':'2018-04-01'],
                     japan['japanStock_ExRet'].loc['2008-07-01':'2018-04-01'],
```

```
                          mexico['mexicoStock_ExRet'].loc['2008-07-01':'2018-04-01'],
                          netherland['netherlandStock_ExRet'].loc['2008-07-01':'2018-04-01'],
                          poland['polandStock_ExRet'].loc['2008-07-01':'2018-04-01'],
                          portugal['portugalStock_ExRet'].loc['2008-07-01':'2018-04-01'],
                          russia['russiaStock_ExRet'].loc['2008-07-01':'2018-04-01'],
                          southafrica['southafricaStock_ExRet'].loc['2008-07-01':'2018-04-01'],
                          southkorea['southkoreaStock_ExRet'].loc['2008-07-01':'2018-04-01'],
                          spain['spainStock_ExRet'].loc['2008-07-01':'2018-04-01'],
                          sweden['swedenStock_ExRet'].loc['2008-07-01':'2018-04-01'],
                          switzerland['switzerlandStock_ExRet'].loc['2008-07-01':'2018-04-01'],
                          uk['ukStock_ExRet'].loc['2008-07-01':'2018-04-01'],
                          us['usStock_ExRet'].loc['2008-07-01':'2018-04-01']],axis=1)
        In [253]:

        ExRet3y = pd.concat([austriaStock_ExRet3Y.loc['2011-07-01':'2018-04-01'],
                            belgiumStock_ExRet3Y.loc['2011-07-01':'2018-04-01'],
                            canadaStock_ExRet3Y.loc['2011-07-01':'2018-04-01'],
                            chechrepublicStock_ExRet3Y.loc['2011-07-01':'2018-04-01'],
                            chileStock_ExRet3Y.loc['2011-07-01':'2018-04-01'],
                            chinaStock_ExRet3Y.loc['2011-07-01':'2018-04-01'],
                            denmarkStock_ExRet3Y.loc['2011-07-01':'2018-04-01'],
                            franceStock_ExRet3Y.loc['2011-07-01':'2018-04-01'],
                            germanyStock_ExRet3Y.loc['2011-07-01':'2018-04-01'],
                            hungaryStock_ExRet3Y.loc['2011-07-01':'2018-04-01'],
                            isrealStock_ExRet3Y.loc['2011-07-01':'2018-04-01'],
                            italyStock_ExRet3Y.loc['2011-07-01':'2018-04-01'],
                            japanStock_ExRet3Y.loc['2011-07-01':'2018-04-01'],
                            mexicoStock_ExRet3Y.loc['2011-07-01':'2018-04-01'],
                            netherlandStock_ExRet3Y.loc['2011-07-01':'2018-04-01'],
                            polandStock_ExRet3Y.loc['2011-07-01':'2018-04-01'],
                            portugalStock_ExRet3Y.loc['2011-07-01':'2018-04-01'],
                            russiaStock_ExRet3Y.loc['2011-07-01':'2018-04-01'],
                            southafricaStock_ExRet3Y.loc['2011-07-01':'2018-04-01'],
                            southkoreaStock_ExRet3Y.loc['2011-07-01':'2018-04-01'],
```

```
                          spainStock_ExRet3Y.loc['2011-07-01':'2018-04-01'],
                          swedenStock_ExRet3Y.loc['2011-07-01':'2018-04-01'],
                          switzerlandStock_ExRet3Y.loc['2011-07-01':'2018-04-01'],
                          ukStock_ExRet3Y.loc['2011-07-01':'2018-04-01'],
                          usStock_ExRet3Y.loc['2011-07-01':'2018-04-01']] ,axis=1)
In [254]:

#Compute the cumulative sum of excess returns
TVERportfolio = np.log(ExRet3y).diff().diff().dropna()*12
TVERportfolio = pd.DataFrame(TVERportfolio)
In [ ]:

plt.figure(figsize=(20, 10))
plt.plot(TVERportfolio, alpha=20);
plt.xlabel('date')
plt.ylabel('returns')
In [ ]:

TVERportfolio.skew()
In [ ]:

TVERportfolio.kurtosis()
In [258]:

ExRreturns_annual = TVERportfolio.mean()*-100
ExRcov_annual = TVERportfolio.cov()*100
In [259]:

# empty lists to store returns, volatility and weights of imiginary portfolios
EXport_returns = []
EXport_volatility = []
EXsharpe_ratio = []
EXstock_weights = []
In [260]:

# set the number of combinations for imaginary portfolios
ExRnum_assets = 25
ExRnum_portfolios = 50000
In [261]:

#set random seed for reproduction's sake
np.random.seed(101)

# populate the empty lists with each portfolios returns,risk and weights
for single_portfolio in range(ExRnum_portfolios):
    weights = np.random.random(ExRnum_assets)
    weights /= np.sum(weights)
    returns = np.dot(weights, ExRreturns_annual)
    volatility = np.sqrt(np.dot(weights.T, np.dot(ExRcov_annual, weights)))
    sharpe = returns/ volatility
    EXsharpe_ratio.append(sharpe)
    EXport_returns.append(returns)
    EXport_volatility.append(volatility)
    EXstock_weights.append(weights)
In [262]:

# a dictionary for Returns and Risk values of each portfolio
portfolio = {'Returns': EXport_returns,
             'Volatility': EXport_volatility,
```

```
                        'Sharpe Ratio': EXsharpe_ratio}

    # extend original dictionary to accomodate each ticker and weight in the portfolio
    for counter,symbol in enumerate(TVERportfolio):
        portfolio[symbol+' Weight'] = [Weight[counter] for Weight in EXstock_weights]
    In [263]:

    # make a nice dataframe of the extended dictionary
    df2 = pd.DataFrame(portfolio)
    In [264]:

    # get better labels for desired arrangement of columns
    column_order = ['Returns', 'Volatility', 'Sharpe Ratio'] + [stock+' Weight' for stock in TVERportfolio]
    In [ ]:

    # reorder dataframe columns
    df2 = df2[column_order]
    df2.head()
    In [ ]:

    # reorder dataframe columns
    df2 = df2[column_order]
    df2.head()
    In [267]:

    # find min Volatility & max sharpe values in the dataframe (df)
    min_volatility2 = df2['Volatility'].min()
    max_sharpe2 = df2['Sharpe Ratio'].max()
    In [268]:

    # use the min, max values to locate and create the two special portfolios
    sharpe_portfolio2 = df2.loc[df2['Sharpe Ratio'] == max_sharpe2]
    min_variance_port2 = df2.loc[df2['Volatility'] == min_volatility2]
    In [ ]:

    # plot frontier, max sharpe & min Volatility values with a scatterplot
    plt.style.use('seaborn-dark')
    df2.plot.scatter(x='Volatility', y='Returns', c='Sharpe Ratio',
                    cmap='RdYlGn', edgecolors='black', figsize=(10, 8), grid=True)
    plt.scatter(x=sharpe_portfolio2['Volatility'], y=sharpe_portfolio2['Returns'], c='red', marker='D', s=200)
    plt.scatter(x=min_variance_port2['Volatility'], y=min_variance_port2['Returns'], c='blue', marker='D', s=200 )
    plt.xlabel('Volatility (Std. Deviation)')
    plt.ylabel('Expected Returns')
    plt.title('Efficient Frontier')
    plt.show()
    In [ ]:

    # print the details of the 2 special portfolios
    print(min_variance_port2.T)
    In [ ]:

    print(sharpe_portfolio2.T)
```