Strategic Asset Allocation

Estimation, Optimization, Visualization

March 2021

Juraj Hledik

National Bank of Slovakia

NATIONAL BANK OF SLOVAKIA
EUROSYSTEM

Preface

This report summarizes my suggestions regarding the process of strategic asset allocation within the National Bank of Slovakia. I have worked on this project during spring and summer of 2022 as a full-time member of the NBS's research department. In case of inquiries regarding the code or the documentation, please contact me at hledik.juraj@gmail.com

Juraj Hledik

Contents

List of Figures

List of Tables

Introduction

This report summarizes key concepts and ideas related to NBS's strategic asset allocation. As is common in the literature on portfolio allocation, we shall divide the process of creating an optimal portfolio according to investor preferences into two consecutive steps:

- Estimation of future asset returns.
- Solving a portfolio optimization problem given the prediction from step 1.

To make our approach as modular as possible, we will threat these two tasks independently, allowing the future users of our pipeline to use them separately. In other words, if someone were to have their own prediction regarding future returns, they are welcome to plug this prediction into our optimization module and use our framework solely for finding an efficient portfolio allocation. Likewise, if one were to have their own portfolio optimization algorithm prepared, they are welcome to use the first part of our approach only (generate future return predictions using our framework) and then plug the predicted returns into their own optimization algorithm. Keeping these two concepts independent therefore gives us further flexibility and freedom in model choice.

There are 4 chapters in this report. Chapter 1 describes estimation of future asset returns from historical data. Chapter 2 then uses this prediction to formulate a modular and flexible portfolio allocation strategy. Finally, Chapter 3 summarizes the graphical user interface used to achieve tasks from previous chapters, describes the data pipeline structure and other technical details.

1
Return estimation

This chapter summarizes the first step in strategic asset allocation - estimation of future returns' probability density function. We will cover everything from data import to the drawing/simulation of future returns from an estimated distribution.

1.1. The data

The data available at our disposal consist of daily returns dating back almost two decades for eight representative asset classes. In the spirit of our modular approach, the investor is able to input their own return observations and preferred asset classes into our framework. Throughout this chapter, however, we will assume a given set of eight asset classes and explain our future return estimation mechanics on this particular asset class set.

The assets used in our analysis contain indices following the prices of gold, US Treasury bonds, a specific ASW portfolio, world equity, Chinese government bonds, US Inflation-linked government bonds, Mortgage-backed securities and emerging markets. These are all summarized in Table 1.1 and form our basic 8 available asset classes.

Asset class	Relevant index
Gold	Gold XAU/EUR Rate FX Unhdg.
US	1-5 Year US Govt (ICE) FX Unhdg.
ASW	1-5 Year Global Non-Sov (ICE) IR hdg. IRS
Stocks	World Equity (MSCI) FX Unhdg.
China	1-3 Year China Govt (ICE) FX Unhdg.
InflBonds	US Inflation-Linked Govt (ICE) FX Hdg.
MBS	US MBS (ICE) FX Hdg.
EM	EM global IG Govt (JPM) FX Hdg.

Table 1.1: Asset classes.

1.2. The return simulation process

Following consecutive steps are part of the return estimation process:

1. User parameters specified. These include the sample size of simulated future return distribution, maturity of said returns, frequency of historical returns used in such estimation etc. For a full list of relevant user parameters, see Table 1.2.
2. Imported daily data.
3. Created a new dataset with observation frequency chosen by the user - daily, weekly, monthly or yearly. Data saved in a relevant folder.
4. Fit each asset class with both Gaussian as well as Pearson type VII distribution.
5. Transform each asset class into a $[0, 1]$ uniformly distributed variable using the distributional assumption of a Pearson type VII distribution.
6. Fit the copula to the uniformly distributed marginal data, drawing from the allowed copula families selected by the user.
7. Use the fitted model to generate $N\_simulation$ observations of simulated returns / draw them from the estimated distibution. Also draw the same number of returns from multidimensional Gaussian distribution.
8. Plot pairwise correlations of the simulated Pearson + Copula vs. Gaussian returns vs. historical data to allow for a visual comparison.

Parameter	Description	Default value
N_simulation	Number of returns that are being simulated. From these, the final returns are drawn uniformly randomly, depending on sample size. Therefore must be greater than $\max$[sample_sizes].	1 000 000
sample_sizes	Sample sizes for which future returns are simulated.	(100, 500, 1000, 2000, 5000, 10000, 50000, 100000)
maturities	Maturities for which future returns are simulated.	(1,3,6,12,60)
frequency	Frequency to which daily historical returns are being transformed first.	monthly
selected_copula_types	Selection of available copula families that are allowed to be used within the model selection procedure.	all copulas available in the VineCopula package for R

Table 1.2: User parameters used in return estimation and simulation.

9. Compute the Cramer and Kolmogorov-Smirnov tests on distributional equality to evaluate the similarity of simulated return distribution to the original distribution.
10. Use the simulated returns to create datasets with different maturities and sample sizes - daily, weekly, monthly and yearly. Save them in a pre-specified directory structure together with the relevant estimated copula structure.

This process can also be schematically viewed as an algorithm in Algorithm 1. Several steps in this procedure however warrant further clarification and details, which we shall focus on in the following paragraphs.

The data on historical returns is stored as a *historical_daily_returns.csv* file with columns corresponding to the different asset classes and rows corresponding to daily returns. Furthermore, we also use an input file called *portfolio_constraints.xlsx* which besides relevant default investor constraints also contains short versions of the respective asset class names.

In the first step, the returns are loaded into R, formatted as numeric values and transformed to a desired frequency chosen by the user. The transformation from daily observations to e.g. weekly observations is done by:

1. adding 1 to each observation,
2. multiplying the observations corresponding to a particular week,
3. subtracting 1 from the result.

The new dataframe is now taken as a baseline. We assign custom asset class names to all variables (taken from *portfolio_constraints.xlsx*) and save this dataset in an appropriate data folder (for directory structure of the project, refer to the technical summary at the end of this report).

In the second step, we estimate both the Pearson and Gaussian marginal distributions that are the best fits for our data. A one-dimensional Gaussian distribution has the following density function:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \tag{1.1}$$

where μ is the expected value and σ is the variance of the distribution. We compute these quantities from the sample for each asset class and hence obtain the desired Gaussian distribution parameters.

A Pearson type VII distribution density is slightly more complicated, with three parameters instead of two:

Data: historical returns
Input: N_simulation, sample sizes, maturities, frequency, copula_types
Result: simulated returns
import historical return data;
if *frequency=="daily"* **then**
  no aggregation;
else
  if *frequency=="weekly"* **then**
    aggregate to weekly returns;
  else
    if *frequency=="monthly"* **then**
      aggregate to monthly returns;
    end
  end
end
set custom asset names;
create directory structure;
save aggregated data;
foreach *asset class* **do**
  estimate the Pearson and Gaussian distribution parameters;
  generate simulated returns from Pearson and Gaussian distribution;
  plot marginal historical return distribution with Pearson and Gaussian fit;
  transform asset class to a $[0, 1]$ uniform distribution;
end
fit a multivariate Gaussian distribution to the data; fit the copula to the uniform data, drawing from allowed copula types;
plot historical vs. simulated returns in a pairwise correlation plot;
run a Cramer test on equality of distributions with the historical data for both models (Gaussian and copula);
foreach *asset class* **do**
  run a Kolmogorov-Smirnov test of distributional equality with the historical data for both models;
end
save results of Cramer and K-S tests;
for *sample sizes* **do**
  for *maturities* **do**
    store the simulated returns in the relevant path; acording to the frequency, compute the p.a. returns and store them in the appropriate path;
  end
end
store the copula structure;

Algorithm 1: Return estimation and simulation.

$$f(x \mid \lambda, \alpha, m) = \frac{1}{\alpha \, \mathrm{B}\left(m - \frac{1}{2}, \frac{1}{2}\right)} \left[1 + \left(\frac{x - \lambda}{\alpha}\right)^2\right]^{-m} \tag{1.2}$$

The benefits associated with this distribution mostly lie in its scalability with respect to its fourth moment - kurtosis. This freedom allows us to better approximate the historical returns not only in their mean and variance, but also in terms of their tails. This property is especially crucial if we are dealing with investor that places high importance on tail-relevant risk measures such as Value-at-Risk or expected shortfall. In light of such preferences, having a reliable way of predicting tail events is of utmost importance.

An alternative parameterization of the Pearson type VII distribution can be achieved by substituting $\alpha := \sigma\sqrt{2m-3}$. In such case, letting m approach the infinity provides us with a well-known Gaussian distribution as the kurtosis approaches the value of 3. Assuming $m > 5/2$ guarantees the existence of the distributions' first four moments. Another substitution of $\lambda = \mu, \alpha = \sqrt{\nu\sigma^2}, m = \frac{\nu+1}{2}$ then gives us the non-standardized Student's t distribution

$$f(x \mid \mu, \sigma^2, \nu) = \frac{1}{\sqrt{\nu\sigma^2}\, \mathrm{B}\left(\frac{\nu}{2}, \frac{1}{2}\right)} \left(1 + \frac{1}{\nu} \frac{(x-\mu)^2}{\sigma^2}\right)^{-\frac{\nu+1}{2}} \tag{1.3}$$

After some basic algebraic computations, we can show that a random variable X with this particular distribution has its first four moments given by the following formulae:

$$\begin{aligned} \text{E}[X] &= \mu && (1.4)\\ \text{Var}[X] &= \sigma^2 \frac{\nu}{\nu-2} && (1.5)\\ \text{Skew}[X] &= 0 && (1.6)\\ \text{Kurt}[X] &= \frac{3\nu-6}{\nu-4} && (1.7) \end{aligned}$$

With this in mind, we can estimate parameters μ, σ and ν from the first four moments of historical returns simply by letting these expressions be equal to sample-specific moments. With the distribution successfully parameterized, we can easily draw an arbitrary number of simulated returns from the estimated distribution.

Figures 1.1 and 1.2 show the relevant fit for both the Gaussian as well as the Pearson (non-standardized Student's t) distribution. The blue line corresponds to the Gaussian distribution, while the red line shows the fit of the Pearson type VII distribution.

We see that - in line with our expectations - using a distribution with variable kurtosis parameter allows for a tighter fit, especially in asset classes with fatter tails. But even though a brief visual examination is a great tool, we will also look into this goodness of fit more formally with proper statistical tests.

Before doing so, however, we shall focus on estimating the whole multidimensional distribution, not just the marginals. Again, we will do this for both the Pearson as well as the Gaussian distribution. In the former case, we shall use vine copulas for mutual pairwise dependencies, while in the latter case, we would estimate parameters of multivariate Gaussian distribution directly.

Starting with the Pearson distribution, we first transform each asset class into a $[0, 1]$ uniform variable which is a necessary first step for fitting a copula model. The process is simply achieved by taking the estimated Pearson distribution function from the previous step and evaluating it at the respective historical return data points. In other words, if the random variable X corresponds to our historical returns with assumed Pearson type VII cumulative distribution function $F(x)$, then the new random variable obtained by:

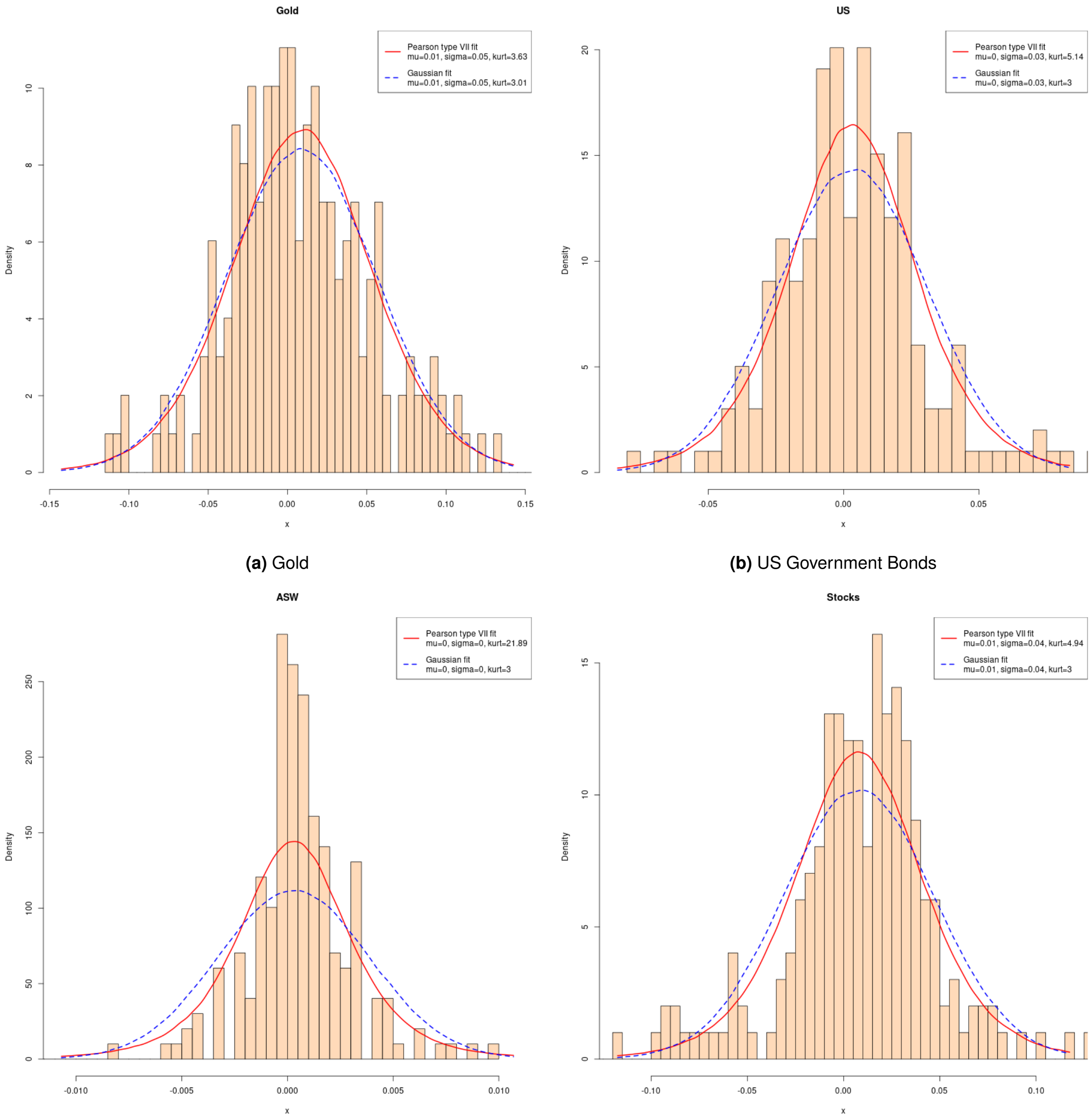
$$U = F(X) \tag{1.8}$$

is uniformly distributed at its domain $[0, 1]$. Transforming all asset classes this way shrinks our 8-dimensional real space into an 8-dimensional unit cube $[0, 1]^8$.

With the data transformed to a unit cube, we take advantage of the R package "Vine Copula", see (Nagler et al., 2021). This package contains functions and routines focused on estimation and model selection of multivariate pairwise copulas also called the vine copulas. The function "*RVineStructureSelect*" allows us to use its heuristical model-selection algorithm which automatically chooses and estimates the best pairwise copula model for our dataset given our choice of allowed copula families. Further details regarding the algorithm and the package in general can be found in (Dißmann et al., 2013).

Once the model is fitted, we draw N_simulation returns from their estimated joint distribution. We follow the same approach with the multivariate Gaussian distribution, only instead of pairwise copulas, we simply use the maximum likelihood approach to estimate the distribution's vector of mean returns μ and the variance-covariance matrix Σ.

At this point, we have three sets of returns at our disposal: historical returns, Pearson type VII distributed returns with their mutual dependency given by pairwise copulas and a return set with multivariate Gaussian distribution. Pairwise dependency plot of these 3 classes of returns can be found in Figure 1.3.



(a) Gold

(b) US Government Bonds

(c) ASW portfolio

(d) World Equity

Figure 1.1: Historical returns, density approximated with a Gaussian and Pearson distributions. Part 1/2.

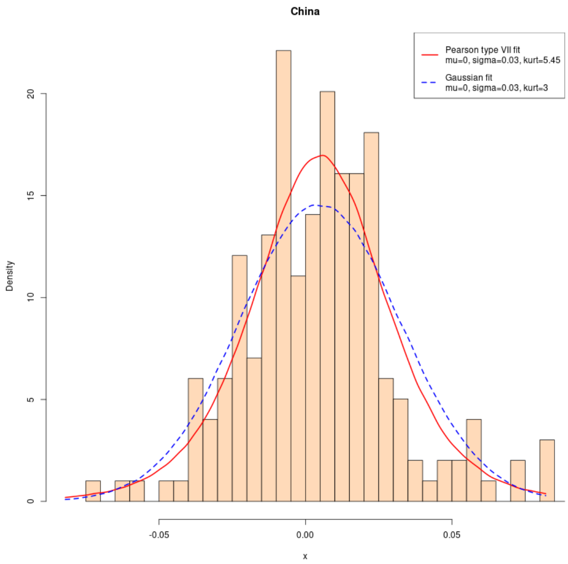
What we can see from the figures directly is that the Pearson-simulated returns are more clustered than the Gaussian-simulated returns, while exhibiting more outliers at the same time. This is due to the Pearson distribution, not the copula structure per se as the Pearson distribution allows us fatter tails than the Gaussian distribution.

In the next step, we examine the similarity between historical and simulated datasets by means of two different statistical tests - the Cramer test an the Kolmogorov-Smirnov test.

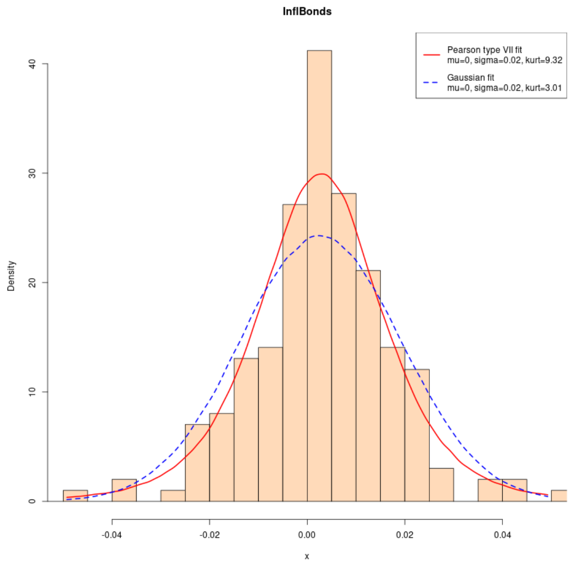
The Cramer test is a multi-dimensional test of distributional equality. It compares two distributions of multiple dimensions with each other and produces a test statistic and p-value corresponding to a null hypothesis $H0$ of equal distributions.

The Kolmogorov-Smirnov test on the other hand only looks at distributions of dimension one. Nevertheless, we can use it on our marginal distributions, so for each asset class separately. Just like in the case of the Cramer test, the null hypothesis $H0$ in the Kolmogorov-Smirnov test also corresponds to the two distributions being equal.

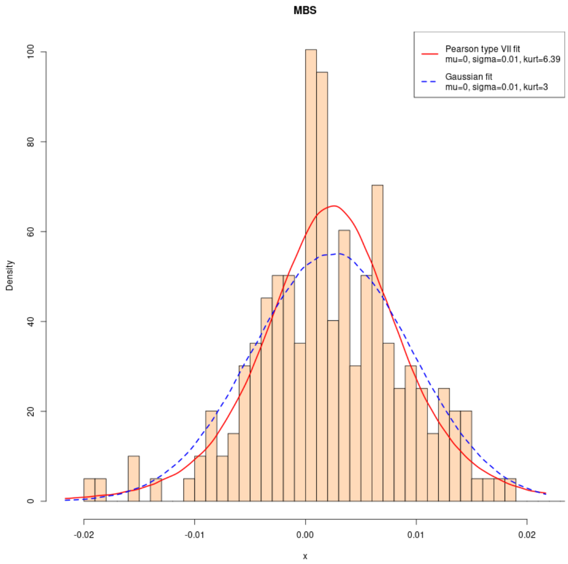
In both tests, we will compare the simulated data with the historical data. We will do this separately



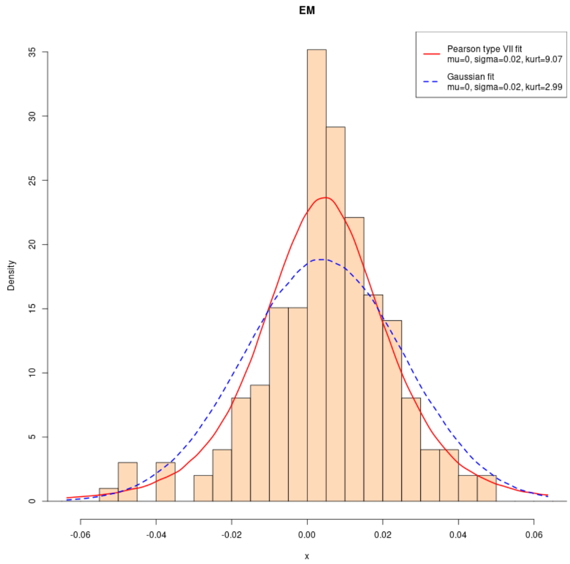
(a) China Government Bonds



(b) US Inflation-linked Government Bonds



(c) US Mortgage-backed Securities



(d) Emerging markets

Figure 1.2: Historical returns, density approximated with a Gaussian and Pearson distributions. Part 2/2

for the Gaussian-simulated returns and for the Pearson-simulated returns. Results for both tests can be found in Table 1.3.

The Cramer test for distributional equality between historical and Pearson-simulated returns does not reject the null hypothesis of equal distributions. In the case of Gaussian distribution, the null hypothesis is rejected in favor of the alternative hypothesis $H1$, therefore showing that generating returns that are Pearson type VII distributed provides more superior results than a simple Gaussian approximation.

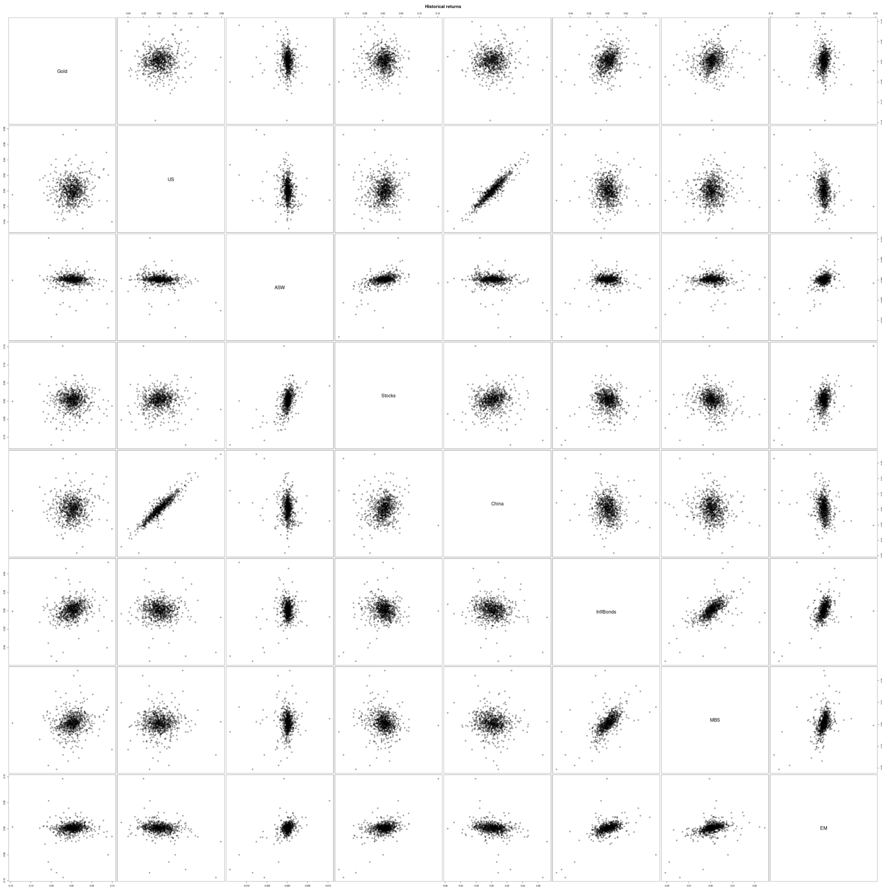
In the case of Kolmogorov-Smirnov test of marginal distributions, the results confirm the null hypothesis in 5 out of 8 asset classes for the Pearson-generated returns and only 1 out of 8 asset classes in the Gaussian-generated returns.

Both tests therefore suggest that using a Pearson type VII distribution allows us to better replicate historical returns than a simple Gaussian approximation.

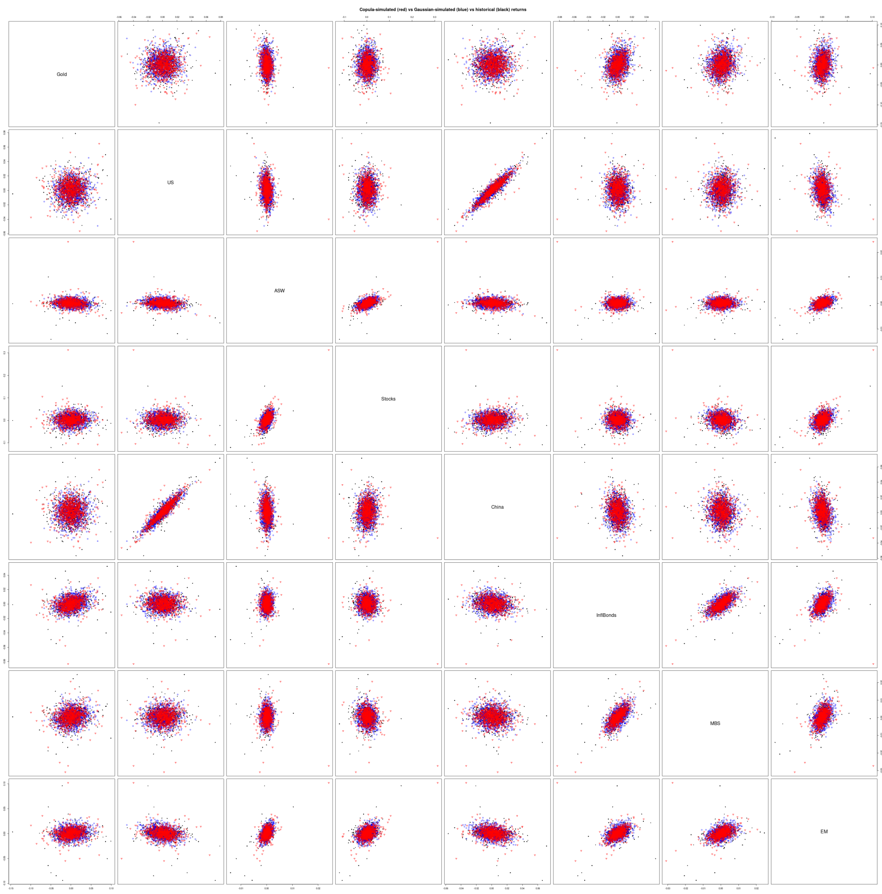
The last step in this part of our algorithm is transforming the returns into the desired set of maturities and sample sizes and storing them in the dedicated folder structure. In this step, we also compute the relevant p.a. returns and store them as well.

Test	Pearson distribution p-value	Gaussian distribution p-value
Cramer test	0.51	0.01
K-S test: Gold	0.50	0.01
K-S test: US	0.00	0.00
K-S test: ASW	0.00	0.00
K-S test: Stocks	0.99	0.12
K-S test: China	0.41	0.00
K-S test: InfBonds	0.06	0.00
K-S test: MBS	0.00	0.00
K-S test: EM	0.79	0.01

Table 1.3: Testing of distributional equality on historical vs. Gaussian and historical vs. Pearson weekly returns.



(a) Historical returns



(b) Historical returns (black), Gaussian-distributed returns (blue) and Pearson-distributed returns(red)

Figure 1.3: Historical returns, density approximated with a Gaussian and Pearson distributions. Part 2/2

Portfolio optimization

In this chapter, we provide methodology for portfolio optimization and its implementation given a modular set of investor constraints.

Throughout the chapter, we shall use multiple different variables. Their description is summarized in Table 2.1.

Variable	Domain	Dimension	Description
K	$\mathbb{N}$	1	Number of asset classes.
N	$\mathbb{N}$	1	Sample size of empirical returns used as prediction of the future.
$\mathbf{r}$	$\mathbb{R}^{N\times K}$	$N \times K$	Empirical return distribution.
$\mathbf{w}$	$[0,1]^{K-1}$	K	Portfolio weights. Need to sum up to 1.
$\mathbf{v}$	$[0,1]^{K-1}$	K	Previous portfolio weights. Need to sum up to 1.
$\bar{\mathbf{x}}$	$\mathbb{R}^{K}$	K	Vector of expected returns for each asset class.
$\mathbf{\Sigma}$	$\mathbb{R}^{K\times K}$	$K \times K$	Variance - covariance matrix of portfolio returns
$\bar{\mathbf{x}}^{\min}$	$\mathbb{R}^{K}$	K	Minimum expected return accepted by the investor.
$\mathbf{w}^{\min}$	$[0,1]^{K}$	K	Vector of minimum allowed portfolio weights.
$\mathbf{w}^{\max}$	$[0,1]^{K}$	K	Vector of maximum allowed portfolio weights.
$\mathbf{a}^{\min}$	$\mathbb{R}^{+K}$	K	Vector of minimum allowed amount for each asset.
$\mathbf{a}^{\max}$	$\mathbb{R}^{+K}$	K	Vector of maximum allowed amount for each asset.
α	$[0,1]$	1	CVaR level. If for instance $\alpha = 0.05$, we look at what happens in 5% of the lowest return realizations.
θ_1	$\mathbb{R}_0^+$	1	Investor importance for the portfolio expected return. The higher the value, the higher the relative importance of expected return in investor's preferences.
θ_2	$\mathbb{R}_0^+$	1	Investor importance for the portfolio variance. The higher the value, the higher the relative importance of variance in investor's preferences.
θ_3	$\mathbb{R}_0^+$	1	Investor importance for the portfolio expected shortfall. The higher the value, the higher the relative importance of expected shortfall in investor's preferences.
θ_4	$\mathbb{R}_0^+$	1	Investor importance for the similarity with the previous portfolio. The higher the value, the lower the willingness of investor to change their portfolio from its previous state.
Ω	$\mathbb{R}^{+}$	1	Total portfolio size.

Table 2.1: Asset classes.

2.1. Objective function

In our approach, the investor can choose how much they care about their portfolio's expected return, variance, expected shortfall and similarity to its last portfolio weights. These quantities' relative importance are expressed in terms of variables $\theta_1, \theta_2, \theta_3$ and θ_4.

The most computationally demanding part of our optimization is looking for the optimal expected shortfall. It can be shown, that if we simply wanted to minimize shortfall, we could write the optimization problem in the following way:

$$
\begin{aligned}
\min_{\mathbf{w},t,\mathbf{z}} \quad -t \quad &- \quad \frac{1}{\lfloor \alpha N \rfloor} \sum_{i=1}^{N} z_i && (2.1)\\
s.t. \quad t + z_i \quad &\leq \quad \mathbf{w}^\top \mathbf{r_i} \quad \forall i \in \{1 \dots, N\} && (2.2)\\
z_i \quad &\leq \quad 0 \qquad \forall i \in \{1 \dots, N\} && (2.3)\\
\mathbf{w}^\top \mathbf{1} \quad &= \quad 1 && (2.4)
\end{aligned}
$$

This result dates back to 1999, see (Rockafellar and Uryasev, 2000) for further details or (Goldberg et al., 2013) for explanation of implementation. Written in this form, the optimization does not prohibit short selling, does not include any minimum allowed expected return, does not consider previous portfolio weights and neither does it include any type of portfolio constraints. It simply and plainly optimizes the expected shortfall by introducing $N+1$ additional variables and $2N$ additional constraints.

In order to adapt this result to our optimization routine, we need to combine it with other optimization objectives - maximizing return, minimizing variance and minimizing difference from our previous portfolio. Depending on relative importance of these concepts, such comprehensive optimization then exhibits the following objective function of $\theta_1, \dots, \theta_4$:

$$
\max_{\mathbf{w},t,\mathbf{z}} \quad \theta_1 \left[\mathbf{w}^\top \bar{\mathbf{x}}\right] + \theta_2 \left[-\frac{1}{2}\mathbf{w}^\top \mathbf{\Sigma} \mathbf{w}\right] \quad + \quad \theta_3 \left[t + \frac{1}{\lfloor \alpha N \rfloor} \sum_{i=1}^{N} z_i\right] + \theta_4 \left[-\frac{1}{2}(\mathbf{w} - v)^\top \mathbf{I} (\mathbf{w} - v)\right] \quad (2.5)
$$

In theory, we could force parameters $\theta_1, \dots, \theta_4$ to satisfy $\theta_1 + \theta_2 + \theta_3 + \theta_4 = 1$. This would turn them into proper weights, truly showing the relative importance of each of the four optimization targets with respect to the rest. It would not change the resulting optimal portfolio since we would be simply multiplying our current objective function by a constant. Nevertheless, it's ok to think about these importance parameters as weights as one can always choose them such that they indeed sum up to 1 if desired.

For computational purposes, this objective function needs to be formulated in the following shape:

$$
\min_{\mathbf{b}} \quad -\mathbf{d}^\top \mathbf{b} + \frac{1}{2}\mathbf{b}^\top \mathbf{D} \mathbf{b} \quad (2.6)
$$

Following steps and substitutions need to be taken to transform this problem from 2.5 to 2.6:

$$\underset{\mathbf{w},t,\mathbf{z}}{\arg\max}\ \theta_1\left[\mathbf{w}^\top\bar{\mathbf{x}}\right]+\theta_2\left[-\frac{1}{2}\mathbf{w}^\top\boldsymbol{\Sigma}\mathbf{w}\right]\ +\ \theta_3\left[t+\frac{1}{\lfloor\alpha N\rfloor}\sum_{i=1}^{N}z_i\right]+\theta_4\left[-\frac{1}{2}(\mathbf{w}-\mathbf{v})^\top\mathbf{I}(\mathbf{w}-\mathbf{v})\right]$$

$$\Updownarrow$$

$$\begin{aligned}\underset{\mathbf{w},t,\mathbf{z}}{\arg\max}\ \theta_1\left[\mathbf{w}^\top\bar{\mathbf{x}}\right]+\theta_2\left[-\frac{1}{2}\mathbf{w}^\top\boldsymbol{\Sigma}\mathbf{w}\right]\ +\ \theta_3\left[t+\frac{1}{\lfloor\alpha N\rfloor}\sum_{i=1}^{N}z_i\right]+&\\ +\theta_4\left[-\frac{1}{2}\mathbf{w}^\top\mathbf{I}\mathbf{w}+\mathbf{w}^\top\mathbf{I}\mathbf{v}-\frac{1}{2}\mathbf{v}^\top\mathbf{I}\mathbf{v}\right]&\end{aligned}\tag{2.7}$$

$$\Updownarrow$$

$$\underset{\mathbf{w},t,\mathbf{z}}{\arg\max}\ \mathbf{w}^\top\left[\theta_1\bar{\mathbf{x}}\right]-\frac{1}{2}\mathbf{w}^\top\left[\theta_2\boldsymbol{\Sigma}\right]\mathbf{w}\ +\ \mathbf{z}^\top\left[\frac{\theta_3}{\lfloor\alpha N\rfloor}\mathbf{1}\right]+t\theta_3-\frac{1}{2}\mathbf{w}^\top\theta_4\mathbf{I}\mathbf{w}+\theta_4\mathbf{w}^\top\mathbf{v}\tag{2.8}$$

$$\Updownarrow$$

$$\underset{\mathbf{w},t,\mathbf{z}}{\arg\max}\ \mathbf{w}^\top\left[\theta_1\bar{\mathbf{x}}+\theta_4\mathbf{v}\right]\ +\ \mathbf{z}^\top\left[\frac{\theta_3}{\lfloor\alpha N\rfloor}\mathbf{1}\right]+t\theta_3-\frac{1}{2}\mathbf{w}^\top\left[\theta_2\boldsymbol{\Sigma}+\theta_4\mathbf{I}\right]\mathbf{w}\tag{2.9}$$

$$\Updownarrow$$

$$\underset{\mathbf{w},t,\mathbf{z}}{\arg\min}\ -\left[\theta_1\bar{\mathbf{x}}^\top+\theta_4\mathbf{v}^\top,\frac{\theta_3}{\lfloor\alpha N\rfloor}\mathbf{1}^\top,\theta_3\right]\begin{bmatrix}\mathbf{w}\\ \mathbf{z}\\ t\end{bmatrix}\ +\ \frac{1}{2}\begin{bmatrix}\mathbf{w}\\ \mathbf{z}\\ t\end{bmatrix}^\top\begin{bmatrix}\theta_2\boldsymbol{\Sigma}+\theta_4\mathbf{I} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{w}\\ \mathbf{z}\\ t\end{bmatrix}\tag{2.10}$$

$$\text{substitution}\ \Bigg\Updownarrow\quad \begin{aligned}\mathbf{d}&:=\left[\theta_1\bar{\mathbf{x}}^\top+\theta_4\mathbf{v}^\top,\tfrac{\theta_3}{\lfloor\alpha N\rfloor}\mathbf{1}^\top,\theta_3\right]\\ \mathbf{D}&:=\begin{bmatrix}\theta_2\boldsymbol{\Sigma}+\theta_4\mathbf{I} & \mathbf{0} & 0\\ \mathbf{0} & \mathbf{0} & 0\\ 0 & 0 & 0\end{bmatrix}\\ \mathbf{b}&:=[\mathbf{w},\mathbf{z},t]\end{aligned}\tag{2.11}$$

$$\underset{\mathbf{b}}{\arg\min}\ -\mathbf{d}^\top\mathbf{b}\ +\ \frac{1}{2}\mathbf{b}^\top\mathbf{D}\mathbf{b}\tag{2.12}$$

As we can see, our objective function can easily be transformed to this canonical quadratic form notation.

2.2. Constraints

In addition to the objective function, we also need to specify the optimization constraints. As we strive to formulate our model as flexible as possible, we shall introduce these constraints in the form of following interchangeable modules:

- Mandatory constraints - equations M1a and M1b,
- minimum expected return - equation M2,
- bounds on portfolio weights - equations M3a and M3b,
- bounds on portfolio amounts if portfolio size Ω is fixed - equations M4Fa and M4Fb,
- equality constraints on portfolio amounts if portfolio size Ω is variable - equation M4V.

2.2.1. Mandatory constraints

The first module contains constraints that are vital for the optimization and always present in any optimization specification. These include the standard constraint on weights summing up to one and a constraint which prohibits short positions in the portfolio:

$$\mathbf{w}^\top\mathbf{1}=1\tag{M1a}$$
$$\mathbf{w}\geq\mathbf{0}\tag{M1b}$$

2.2.2. Minimum expected return

The second module is optional and allows investor to demand a given minimum expected portfolio return:

$$\mathbf{w}^\top \bar{\mathbf{x}} \geq \bar{\mathbf{x}}^{\min} \tag{M2}$$

2.2.3. Limits on portfolio weights

The third module contains upper and lower bounds on individual weights. These allow us to forcefully contain a minimum or a maximum proportion of total portfolio in any given asset class. If $w_i^{\min} = w_i^{\max}$ for some element $i \in \{1, \ldots, K\}$, then asset i's proportion in the portfolio is imposed directly:

$$\mathbf{w} \geq \mathbf{w}^{\min} \tag{M3a}$$

$$\mathbf{w} \leq \mathbf{w}^{\max} \tag{M3b}$$

2.2.4. Limits on portfolio amounts for fixed Ω

In the fourth module, rather than restricting the individual asset class weights, we restrict the total amount (volume) allowed in each asset class. This module is only available if the overall portfolio size Ω is fixed:

$$\mathbf{w} \geq \frac{\mathbf{a}^{\min}}{\Omega} \tag{M4Fa}$$

$$\mathbf{w} \leq \frac{\mathbf{a}^{\max}}{\Omega} \tag{M4Fb}$$

2.2.5. Limits on portfolio amounts for variable Ω

If the overall portfolio size Ω is also a control variable in our optimization, we unfortunately cannot include any inequality constraints. The reason behind this is that they would turn our linear-quadratic problem into a more general non-linear problem. Such change would in turn force us to use a more general optimization algorithm and therefore increase the overall computational complexity of the problem. In such cases, investors are advised to focus on relative constraints on w addressed above instead.

Nevertheless, we can still work with equality constraints where $a_i^{\min} = a_i^{\max}$ for some element $i \in \{1, \ldots, K\}$. Therefore, if the amount for some asset classes is supposed to be fixed, we can still achieve that even with variable Ω. Suppose there are K_{eq} asset classes for which $a_i^{\min} = a_i^{\max}$ such that their amount in the portfolio is fixed and suppose that we denote the set of their indices shall by S_{eq}. Then for each asset $i \in S_{\text{eq}}$, it holds that:

$$w_i = \frac{a_i^{\min}}{\Omega} = \frac{a_i^{\max}}{\Omega} \quad \forall i \in S_{\text{eq}} \tag{2.13}$$

Solving for Ω gives us:

$$\Omega = \frac{a_i^{\min}}{w_i} = \frac{a_i^{\max}}{w_i} \quad \forall i \in S_{\text{eq}} \tag{2.14}$$

Therefore, the following has to hold for every pair of asset classes with fixed portfolio amounts:

$$\frac{a_i^{\min}}{w_i} = \frac{a_j^{\min}}{w_j} \quad \forall i, j \in S_{\text{eq}} \tag{2.15}$$

Reshuffling this equation provides us with the next constraint module, exclusive for a setting with variable portfolio size Ω:

$$w_i a_j^{\min} - w_j a_i^{\min} = 0 \quad \forall i, j \in \{1, \ldots, N\} : (a_i^{\min} = a_i^{\max}) \cap (a_j^{\min} = a_j^{\max}) \tag{M4V}$$

For more than two fixed amount asset classes, some of these equality constraints are obviously redundant which we take care of in our implementation.

The bottom line for determining optimal portfolio size with variable Ω is as follows:

- If no asset classes amounts are fixed, such that $\forall i \in \{1, \ldots, N\}$ it holds that $a_i^{\min} \neq a_i^{\max}$, then the overall optimal portfolio size Ω is set equal to the previous portfolio size.
- If a single asset class i's amount is fixed, we first compute the optimal portfolio weights vector $\mathbf{w}^\star$ without any constraints involving Ω and then determine the optimal Ω from this one fixed amount constraint by setting

$$\Omega := \frac{a_i^{max}}{w_i^\star}. \tag{2.16}$$

- If multiple asset classes have their desired amounts in the portfolio fixed, we include additional constraints according to Equation M4V, compute the optimal weights vector $\mathbf{w}^\star$ and then evaluate the overall portfolio size Ω exactly as in the previous case with i being one of the asset classes present in the equality constraints.

Bibliography

J. Dißmann, E. C. Brechmann, C. Czado, and D. Kurowicka. Selecting and estimating regular vine copulae and application to financial returns. *Computational Statistics & Data Analysis*, 2013.

Lisa R. Goldberg, Michael Y. Hayes, and Ola Mahmoud. Minimizing shortfall. 2013.

Thomas Nagler, Ulf Schepsmeier, Jakob Stoeber, and Eike Christian Brechmann. *Package 'VineCopula'*, October 2021.

R. Tyrrell Rockafellar and Stanislav Uryasev. Optimization of conditional value-at-risk. *Journal of Risk*, 2000.