Technische Universiteit Delft
Faculteit Elektrotechniek, Wiskunde en Informatica
Delft Institute of Applied Mathematics

Robuuste schattingen in de financiële sector

(Engelse titel: Robust estimation in the financial sector)

Verslag ten behoeve van het
Delft Institute of Applied Mathematics
als onderdeel ter verkrijging

van de graad van

BACHELOR OF SCIENCE
in
TECHNISCHE WISKUNDE

door ESMEE VERMOLEN

Delft, Nederland
November 2017

BSc verslag TECHNISCHE WISKUNDE

“Robuuste schattingen in de financiële sector”

(Engelse titel: “Robust estimation in the financial sector”)

ESMEE VERMOLEN

Technische Universiteit Delft

Begeleider

Dr.ir. L.E, Meester

Overige commissieleden

Dr.ir. F.H. van der Meulen Dr. J.G. Spandaw

November, 2017 Delft

Abstract

Risk estimations play an important role in the financial sector. This report focuses on the risk measure Value at Risk, which represents the risk of a portfolio. We will analyse different estimators for the standard deviation and correlation and specifically look at their resistance to outliers, also called the robustness of the estimator.

We will first explain why risk estimation is used in the financial sector, what the Value at Risk is and how we define robustness. The different standard deviation estimators that will be used in simulation are analysed and we will first test their robustness for data drawn from location-scale families. We will compare their sampling distributions for unpolluted data sets from the normal and logistic distribution to data sets from the same distributions that contain an extreme outlier. We will also simulate the Value at Risk estimation with the different standard deviation estimators with three real price data sets.

The Value at Risk estimation for portfolios of 2 assets are based on the standard deviation estimations for the separate assets and on the correlation between the two assets. The correlation estimators will also be analysed using data with and without an outlier, drawn from the normal and logistic distribution. We also look into the Factor model, a different way to model price data. Last, a simulation for a portfolio of 2 assets with real data will show us how the correlation and standard deviation estimators estimate the Value at Risk.

Contents

Chapter 1

Introduction

Risk is a concept that plays an essential role in the financial sector, but in order to manage these risks it is useful to first understand what risk is and how it is calculated. An important realisation is to know that risk is not a given, it is an estimation. There are many different ways to estimate different kinds of risks. This project will concentrate on the estimations of the Value at Risk (VaR), which is a commonly used risk measure to represent the risk of a portfolio. We want to see how different estimators react to anomalies and focus on their resistance to outliers. This is the robustness of the estimator.

I first encountered the terms Value at Risk and robust during my summer internship at Transtrend. During this internship I looked at different estimators for portfolios of 1 asset and analysed their behaviour and robustness by simulation. The price data used in this project is provided by Transtrend. This project will expand to portfolios of 2 assets and will focus more on the theoretical aspects. Expanding to two-dimensional problems adds complexity to the risk measure because correlation will now play a role as well as the volatility of the assets.

Before we will look into the different estimators for the standard deviation and correlation in the next chapter, we will introduce some concepts that are important to this project: Risk estimation, Value at Risk, and robustness. We will look into VaR estimations for portfolios of 1 asset in chapter 3 and discuss the performance of the estimators in VaR calculations, using portfolios constructed with real price data. In chapter 4 we will look into VaR estimations for portfolios of 2 assets and simulate them with real price data.

1.1 Risk estimation in finance

Risk estimations are used for business decisions such as selling of or investing in assets. One of the reasons why accurately measuring (investment) risk is very important is the relationship between the return of an investment and its risk. The higher the risk, the higher the expected return and visa versa. At the same time, the higher the risk, the more you could potentially lose from an investment. So investors also want to measure, as accurately as possible, how much value of their investment is at risk. The most commonly used measure for this is the Value at Risk (VaR).

Besides the influence on business decisions, risk estimations are also used for risk management. There are rules both within companies and on an European level that put limits on certain risks. In this report we will focus on the risks of portfolios and a specific rule from the European Central Bank (ECB) that applies to this. The ECB uses the VaR measure to set a limit on the risks of portfolios. As soon as the VaR of a portfolio exceeds this limit, the investors are forced to sell part of their holdings in these assets to lower the VaR. So the accuracy of the

VaR measure is important to both the investor and the market. If this VaR estimation exceeds the limit, the investor is forced to sell this asset which influences the market. The VaR measure can be calculated in many different ways, all leading to different outcomes. Because of the rules of the ECB, a different outcome could have big impact. In the next section we will look into the VaR measure.

1.2 Value at Risk

Mathematically, the VaR is a quantile of the distribution. This means that to calculate the VaR, it is important to know the volatility of your portfolio. If L is the price of a portfolio, the VaR given level α is:

$$VaR_\alpha = \inf\{l \in \mathbb{R} : \mathbb{P}(L > l) \leq 1 - \alpha\}.$$

The VaR is widely used and it is a practical risk estimate, as long as it gives an accurate estimation. However, it has some disadvantages when the estimation is inaccurate. Clearly, underestimating the risk of an investment is undesirable and could be dangerous for the investor. On the other hand, overestimating the risk of an investment also causes some trouble, as we have seen in the previous section. The investors have to sell their investments if the VaR exceeds the limit set by the ECB. This has a big impact on the entire market because most investors use the same methods to calculate the VaR.

When dealing with actual price data, the volatility of a portfolio will be estimated using estimators, so the accuracy of the VaR will heavily depend on the performance of the estimators. The problem with some estimators is that they work very well under normal circumstances but overestimate the volatility in the case of extreme market movements. This means that if an unexpected event occurs, the estimator will overestimate the volatility of the portfolio and the VaR estimate will unnecessarily explode. The (in)sensitivity to outliers is called the robustness of an estimator, which we will discuss in the next section.

1.3 Robustness

Robustness has been defined in many different ways, we will use the definition used by Peter J. Huber: "Robustness signifies insensitivity to small deviations from the assumptions". [1]

The following simplified example will show the differences in robustness for two location estimators. We have the set $X = (1, 2, 4, 3, 6, 3)$ and we will compute the median and the sample mean. The median is 3 and the sample mean is $3\frac{1}{6}$. Now, if we add 100 to our set: $X = (1, 2, 4, 3, 6, 3, 100)$, the median is still 3 but the sample mean is now $19\frac{5}{6}$. In order for the median to explode, we would have to add more extreme values, while the mean gives a very different outcome when we add just 1 value to our set. The median is a more robust estimator than the sample mean.

The behaviour of the sample mean from the example is unwanted when an outlier occurs in price data. If the standard deviation estimator explodes from just 1 outlier, the VaR will drop or rise, resulting in an underestimation of the risk or it could overestimate the risk and exceed the VaR limit. However, the risk of this asset might not have changed this much. If the extreme value turns out to be a single outlier, the VaR should not have exploded. When studying the price data of several assets, this event of a single outlier occurs often. We will study the behaviour of different standard deviation estimators and see how they respond to these kind of outliers.

In this report, we will also look at portfolios of two assets. When dealing with two assets, their correlation will also influence the VaR. When anomalies occur, the correlation estimates could increase, decrease or change sign. So it is less intuitive to understand what we want to protect our correlation estimators from. We will look at these different anomalies and study the behaviour of the different estimators and the effect on the VaR.

Chapter 2

Estimators

To measure the VaR, we have to know the volatility of the portfolio. The volatility of the portfolio depends on the standard deviation of the assets in this portfolio and on the correlation between these assets. To determine the standard deviation and correlation, we use estimators. In this chapter we will look into different types of estimators that can be used. We will look at how the different estimators define and react to outliers and we will study the behaviour of the estimators by simulation in the next chapters.

Note that the data we focus on in this report contains price changes and the sample mean of the data sets are close to zero. Therefore, we assume that the mean of the data equals zero. Therefore, we will not look into estimators for the mean. We will only look into different types of estimators for the standard deviation and correlation.

2.1 Standard deviation estimators

For the standard deviation estimators, we will use the categorisation proposed by Huber[1]. In chapter 3 of [Robust Statistics, 2009] Huber classified the M- and L-estimators. We will discuss these types of estimators and give examples of standard deviation estimators for each type.

2.1.1 M-estimators

Before we dive into the general definition of M-estimators, we will look at a familiar example of an M-estimator, the maximum likelihood estimator (MLE). In general the MLE is calculated in the following way. [3]

Suppose that random variables $X_1, ..., X_n$ have a joint density function $f(x_1, x_2, ..., x_n|\theta)$. Given observed values $X_i = x_i$ where $i = 1, ..., n$, the likelihood of θ as a function of $x_1, .., x_n$ is defined as: $lik(\theta) = f(x_1, x_2, ..., x_n|\theta)$. The MLE $\hat{\theta}$ is the value of θ maximizing $lik(\theta)$. To find the MLE, the natural logarithm of the likelihood is often used because it is easier to find the maximum. For i.i.d. samples, the likelihood is $lik(\theta) = \prod_{i=1}^{n} f(x_i|\theta)$ and the log likelihood is $L(\theta) = \sum_{i=1}^{n} \log[f(x_i|\theta)]$

For example, given that $X_1, X_2, ..., X_n$ are i.i.d. $N(0, \sigma^2)$, their joint density function is:

$$f(x_1, x_2, ..., x_n|\sigma^2) = \prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{x_i}{\sigma})^2}$$

And the log likelihood is:

$$L(\sigma) = \sum_{i=1}^{n} \log[\frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{x_i}{\sigma})^2}] = -n\log(\sigma) - \frac{n}{2}\log(2\pi) - \frac{1}{2\sigma^2}\sum_{i=1}^{n} x_i^2$$

To find the MLE, maximise the log likelihood and solve $\frac{\delta L(\sigma)}{\delta \sigma} = 0$. We find

$$\frac{-n}{\sigma} + \frac{\sum (x_i)^2}{\sigma^3} = 0 \Rightarrow \hat{\sigma} = \sqrt{\frac{\sum (x_i)^2}{n}}$$

The problem with the MLE is that it is not resistant to outliers. A single outlier can affect the estimator greatly.

The M-estimators is a group of maximum likelihood type estimators, giving alternatives to the MLE to find a more robust estimator. To find the MLE we have to solve $\sum \frac{\delta}{\delta \theta} log f(x_i; \theta) = 0$, and the general definition of M-estimators is: [1]

$$\sum \frac{\delta}{\delta \theta} \rho(x_i; \theta) = 0 \tag{2.1}$$

Where the function ρ is an arbitrary function.

An example of such a more complex m-estimator, the Huber m-estimator, will be discussed in the next section.

Huber m-estimator

The Huber m-estimator is also a maximum likelihood type estimate but it is constructed to be more resistant to outliers. The Huber m-estimator filters out the outliers before estimating the standard deviation. We will describe this method in this section.[1]
Before estimating the standard deviation for data set $X = (x_1, ..., x_n)$, the Huber m-estimator filters the data set in the following way. Choose a fixed k, initiate with $\mu_0 = \text{median}(X)$ and $\sigma_0 = \text{MAD}(X)$. Define $Y = (y_1, ..., y_n)$ where $y_i, i = 1, ..., n$ is defined as:

$$y_i = \begin{cases} x_i, & \text{for } \mu_0 - k\sigma_0 \leq x_i \leq \mu_0 + k\sigma_0 & \text{(2.2a)} \\ \mu_0 - k\sigma_0, & \text{for } x_i \leq \mu_0 - k\sigma_0 & \text{(2.2b)} \\ \mu_0 + k\sigma_0, & \text{for } x_i \geq \mu_0 + k\sigma_0 & \text{(2.2c)} \end{cases}$$

So by choosing k, you determine when you consider a value to be an outlier. When the set contains an outlier, it will not be included in the calculation. Instead, it will be replaced by either $\mu - k\sigma$ or $\mu + k\sigma$, depending on whether it is an outlier below or above the mean. This is what makes the estimator more robust than the MLE.

However, when k is very small, a lot of values could be replaced which will give a biased estimate for the standard deviation. Huber compensated for this effect in the calculation of the Huber m-estimator with the constant $\beta(k)$. The relation between $\beta(k)$ and k is displayed in figure 2.1. The lower you choose k, the lower β will be, which will increase the σ_{huber} to compensate for the replacement of outliers. The default value for k is 1.5, which we will also use in this project.

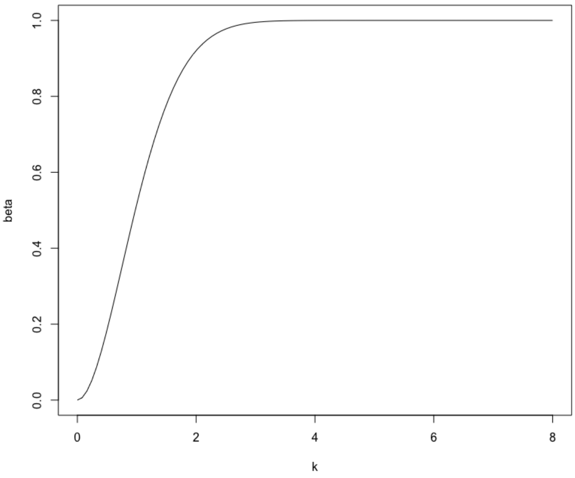


Figure 2.1: β

After replacing the outliers, the estimation of the standard deviation is computed as:

$$\hat{\sigma}_{huber} = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \text{median}(Y))^2}{n-1}\frac{1}{\beta(k)}}$$

The Huber estimator is more resistant to outliers so a single outlier will not affect the estimation of the Huber estimate as much as the sample standard deviation.

2.1.2 L-estimators

L-estimates are linear combinations of order statistics. Since any linear combination of order statistics is an L-estimator, they can be as simple as a single point or the median. This is one of the main benefits of these types of estimators. They are often simple, easy to calculate and to interpret and are often resistant to outliers. This makes it a very useful estimator in robust statistics.

MAD estimator

The L-estimator that we will focus on for the standard deviation is a constant times the median absolute deviation (MAD). The definition of the MAD is: [1]

$$MAD = \text{median}(|x_i - \text{median}(X)|)$$

And the estimator for the standard deviation is given by:

$$\hat{\sigma}_{MAD} = K \cdot MAD$$

The relation between MAD and the estimator for the standard deviation is determined by K, which is a scale factor determined by the distribution to make the estimator consistent. For the normal distribution, $K = \dfrac{1}{\Phi^{-1}(\frac{3}{4})} \approx 1.4826$.

This can be derived from the definition of the MAD, because for $n \to \infty$, $\mathbb{P}(|x - \mu| \leq MAD) = \frac{1}{2}$ for the estimator the be consistent. Which means:

$$\frac{1}{2} = \mathbb{P}(|\frac{x-\mu}{\sigma}| \leq |\frac{MAD}{\sigma}|)$$

$$\Rightarrow \frac{1}{2} = \Phi(\frac{MAD}{\sigma}) - \Phi(\frac{-MAD}{\sigma})$$

$$\Rightarrow \frac{1}{2} = \Phi(\frac{MAD}{\sigma}) - (1 - \Phi(\frac{MAD}{\sigma}))$$

$$\Rightarrow MAD = \Phi^{-1}(\frac{3}{4}) \cdot \sigma =$$

$$\Rightarrow K = \frac{1}{\Phi^{-1}(\frac{3}{4})} \approx 1.4826$$

Interquartile range

Another order statistic that can be used to estimate the standard deviation is the interquartile range (IQR). Quartiles divide a rank-ordered data set in four equal parts. The values that divide the set into these four parts are called the first, second and third quartiles, Q_1, Q_2 and Q_3 respectively, where Q_2 is the median. The interquartile range (IQR) is defined as $IQR = Q_3 - Q_1 = CDF^{-1}(\frac{3}{4}) - CDF^{-1}(\frac{1}{4})$. To estimate the standard deviation, the IQR estimator also needs a scale factor, K, that is determined by the distribution. The IQR estimator for the standard deviation is $\hat{\sigma}_{IQR} = K \cdot IQR$.
For the normal distribution, the scale factor K can be calculated as:

$$\hat{\sigma}_{IQR} = \frac{1}{1.3490} \cdot \text{IQR}$$

Because the $\mathbb{E}(\text{IQR}) = \Phi^{-1}(\frac{3}{4}) - \Phi^{-1}(\frac{1}{4}) = 2 \cdot \Phi^{-1}(\frac{3}{4}) = 1.3490$.

2.2 Correlation estimators

If we have a portfolio consisting of two or more assets the correlation between the different assets influences the value at risk, so we also need to take the correlation between these assets into account. In this section we will discuss different estimators for the correlation. We will not use the same categorisation as we did in the last section. As mentioned in section 3.1 of "Robust estimation"[1], M-estimators are most useful for multiparameter problems. We will discuss the Pearson and Kendall τ correlation estimators, as well as the estimators from the Robust r package[4] that we will use in the simulations to come.

2.2.1 Pearson correlation

The most commonly used estimator for correlation is the Pearson correlation measure. Given the data $x_1, ..., x_n$ and $y_1, ..., y_n$, the Pearson correlation coefficient for the pairs $(x_i, y_i), i = 1, ..., n$ is:[3]

$$\hat{\rho}_{x,y} = \frac{\sum_i (x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_i (x_i - \overline{x})^2 \sum (y_i - \overline{y})^2}}$$

It measures the strength of linear relationship, which can be an undesirable characteristic when dealing with assets since their relationship can also be non-linear. Another problem that might arise when we calculate the correlation with this estimator is that it does not filter outliers which makes it a non robust estimator. One extreme outlier can make the Pearson correlation measure go anywhere between -1 and 1.

2.2.2 Kendall τ correlation

The Kendall τ correlation is a rank estimator, it measures the rank correlation. For $(x_1, y_1), ..., (x_n, y_n)$ a set of observations of random variables X and Y. We call (x_i, y_i) and (x_j, y_j) concordant if $x_i > x_j$ and $y_i > y_j$ or both $x_i < x_j$ and $y_i < y_j$. In the cases $x_i > x_j$ and $y_i < y_j$ or $x_i < x_j$ and $y_i > y_j$ the pair is discordant. If $x_i = x_j$ or $y_i = y_j$ they are neither concordant or discordant. Then, the Kendall τ correlation is defined as: [2]

$$\tau = \frac{\text{nr of concordant pairs - nr of discordant pairs}}{\frac{1}{2}n(n-1)}$$

We will look at a simplified example to see how this estimator works and reacts to an outlier. Given the length and weight of 5 people: $(167, 62), (170, 58), (172, 70), (178, 68), (182, 75)$, the number of concordant pairs is 8 and the number of discordant pairs is 2, so $\tau = \frac{8-2}{\frac{1}{2}\cdot 5\cdot 4} = \frac{3}{5}$. This indicates that there is a strong correlation between the length and weight of a person. If we add the outlier $(150, 70)$, the number of concordant pairs is 9 and the number of discordant pairs is 5, resulting in $\tau = \frac{9-5}{\frac{1}{2}\cdot 5\cdot 4} = \frac{2}{5}$. So the outlier does influence the correlation coefficient, but not the correlation does not change sign in this example. In simulation we will see how this estimator reacts to different types of outliers.

2.2.3 Robust R package

The Robust r package provides many functions related to robust procedures. We wil use the covRob function from this package. As mentioned in Package 'robust'[4]: "The covRob function selects a robust covariance estimator that is likely to provide a good estimate in a reasonable amount of time". In its default setting, the covRob function selects one of the following estimators based on the problem size:
The Donoho-Stahel estimator(SD), (Stahel, 1981 and Donoho, 1982) when there are less than 5,000 oberservations or the Minimum Covariance Determinant (MCD) estimator of Rousseeuw (1985) when there are less than 50,000 but more than 5,000 observations. We will briefly describe each of the estimators that we will use in this report based on [Robust and efficient estimation of multivariate scatter and location][6] and the covRob package. [4]

1. The Donoho-Stahel estimator
   The Donoho-Stahel estimator uses a weight function that assigns weights to data points, based on whether they are an outlier. Given $X = (x_1, .., x_n)$, this estimator defines the outlyingness for $x_i, i = 1, .., n$ and $a \in \mathbb{R}$ as

   $$r(x_i) = max_a \frac{|ax_i - \text{median}(aX)}{\text{MAD}(aX)}.$$

   The Donoho-Stahel estimator will give weights to each data point, based on their outlyingness. The higher the outlyingness, the lower the weight given to that data point. This way, the outliers are filtered. After the weights are assigned to each data, the Donoho-Stahel estimator determines the sample correlation of the weighted data.

2. The Minimum Covariance Determinant estimator
   Instead of looking at the individual elements of the correlation matrix, the MCD estimator looks for the covariance matrix with the smallest determinant by calculating the covariance matrix for all subsets of data points.
   It takes a lot of time to check all possible subsets to find the MCD, so this function uses the Fast MCD algorithm of Rousseeuw and Van Driessen. This algorithm only looks at subsets with m observations out of n, where $\frac{n}{2} < m \leq n$.

The different estimators that the covRob can use, identify different data points as outliers which will result in different outcomes. In the next chapters we will look into the estimators discussed in this chapter and study their behaviour in simulation.

Chapter 3

Portfolio of 1 asset

In this chapter we will see how the different standard deviation estimators perform when they are given data from different distributions. We will first look at general location-scale families, and later look at two members of this family: the normal and logistic distribution. We will then look at the behaviour of the different standard deviation estimators if we assume that the asset follows a model from the location-scale family. At the end of this chapter we will also look at real price data to simulate VaR calculations for portfolios of a single asset.

We will simulate the estimation of the standard deviation as we would with real data. To do this, we have to introduce two concepts: time-series and the lookback window.
The data that we will use are all time-series. "In time-series data, a single individual is tracked over many time periods or points of time" [5]
The lookback window is the number of previous data points the estimator is allowed to use to make its estimation. The more observations they can use, the less a new observation can influence the estimation. On the other hand, too many observations can make that the history of the data influences the estimator so much, it does not give an accurate estimation. We will not constrain the lookback window when we look at data from the location-scale family. When running the simulations with real data sets, we will also look at different lookback windows to see how the lookback window influences the estimations.

3.1 Location-scale family

A location-scale family is a family of distributions that is determined by the location parameter a and scale parameter b. Assume Y is a random variable with density ψ and cumulative distribution function Ψ. Let $X = a + bY$. The density of the location-scale family distribution is of the form:

$$f(x|a,b) = \frac{1}{b}\psi(\frac{x-a}{b}),$$

where ψ is a probability density function, a is the location parameter and b the scale parameter. The CDF of the family is of the form:

$$F(x|a,b) = \Psi(\frac{x-a}{b})$$

From this we can determine the quantile function of X:

$$Q(\alpha) = F^{-1}(\alpha) = a + b\Psi^{-1}(\alpha)$$

Because of our assumption that the mean equals zero, $\mathbb{E}(Y) = 0$, and we want that also $\mathbb{E}(X) = 0$. That means that $\mathbb{E}(X) = a + b\mathbb{E}(Y)$, so we set $a = 0$. Both of the examples that we will discuss, logistic and normal distribution, are symmetric if $\mu = 0$, so for both distributions, the VaR of a portfolio of one asset from the location-scale family is determined by:

$$VaR_\alpha = b \cdot \Psi^{-1}(\alpha) \tag{3.1}$$

3.1.1 Normal distribution

If we define $\psi(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}$, which is the density of the standard normal distribution ϕ, then

$$f(x|\mu,\sigma) = \frac{1}{\sigma}\psi(\frac{x}{\sigma}) = \frac{1}{\sqrt{2\sigma^2\pi}} \cdot e^{\dfrac{-x^2}{2\sigma^2}}$$

is the probability density function for the normal distribution with mean μ and standard deviation σ. From 3.1 follows:

$$VaR_\alpha = \sigma \cdot \Psi^{-1}(\alpha) = \sigma \cdot \Phi^{-1}(\alpha).$$

To simulate the VaR estimations, we have to know σ. We will use the estimators discussed in the previous chapter, we will evaluate and discuss the following estimators for σ.

1. $\hat{\sigma}_{sd} = \sqrt{\dfrac{\sum(x_i - \hat{\mu})^2}{n}}$
2. $\hat{\sigma}_{huber}$, as described in Section 2.1.1
3. $\hat{\sigma}_{MAD} = \dfrac{1}{\Phi^{-1}(\frac{3}{4})} \cdot MAD = 1.4826 \cdot MAD$
4. $\hat{\sigma}_{IQR} = \dfrac{1}{2\Phi^{-1}(\frac{3}{4})} \cdot IQR = 0.7412 \cdot IQR$

We ran 1,000 simulations, each with 100 observations of the normal distribution with mean 0 and standard deviation 1 to determine the distribution of the estimators. The sampling distribution of the four standard deviation estimators can be seen in figure 3.1.

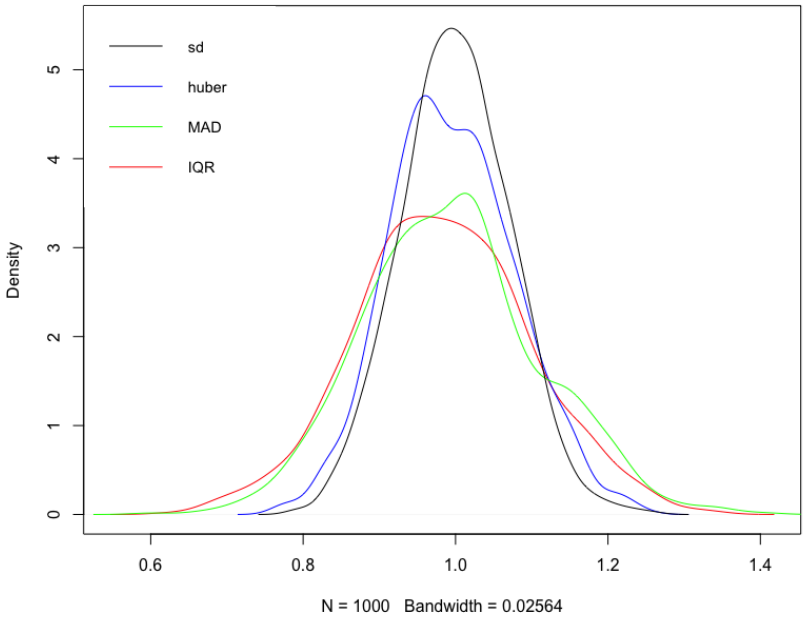


Figure 3.1: Sampling distributions of the standard deviation estimators for normally distributed data

One of the first things that stands out from figure 3.1 is that there is a clear difference between all the estimators. They all have a peak around the same value but have a really different distribution in the tails. The L-estimators, IQR and MAD, have fatter tails and a lower peak then the M-estimators. This means that their estimations vary more, and that for this specific simulation, the Huber and sample standard deviation would be our preferred estimators.

We know from the previous chapter that the IQR, MAD and Huber estimators are more robust than the sample standard deviation. The more robust estimators will have fewer outliers than the sample standard deviation because they do not explode from a single outlier. This means that the tails of the distributions of the more robust estimators will be thinner. This is not what we see in figure 3.1, but that is because these simulations did not have outliers. When we simulate this with real price data, we will see a bigger difference between the different estimators.

To see how the estimators react to anomalies, we have added an outlier to a time-series drawn from the normal distribution. We have drawn a 1,000 times 100 observations from the standard normal distribution and replaced the 60th data point with 8 times the value of the sample standard deviation. Figure 3.2 shows an example of one of the simulations in which we can see the clear outlier at $t = 60$. Figure 3.3 shows the standard deviation estimations from the different estimators and figure 3.4 shows the sampling distribution of the estimators for the 1,000 simulations. We did not constrain the lookback window yet, so at time t, the estimators give an estimation using the data available up until time t.

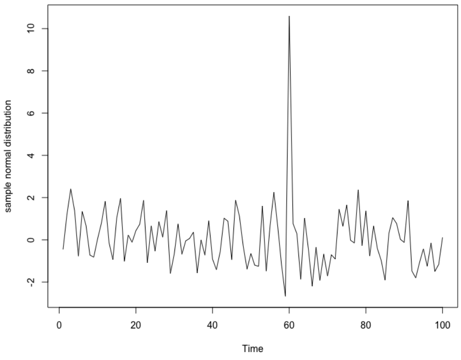


Figure 3.2: Example of a polluted normally distributed data used in the simulation

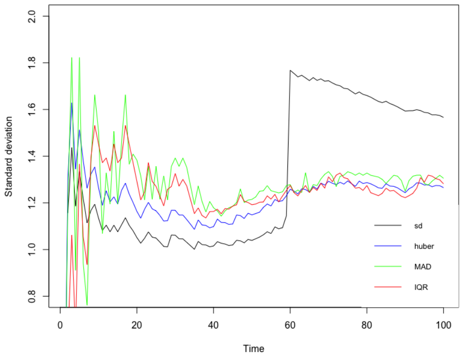


Figure 3.3: Example of the standard deviation estimation given a polluted normally distributed data set

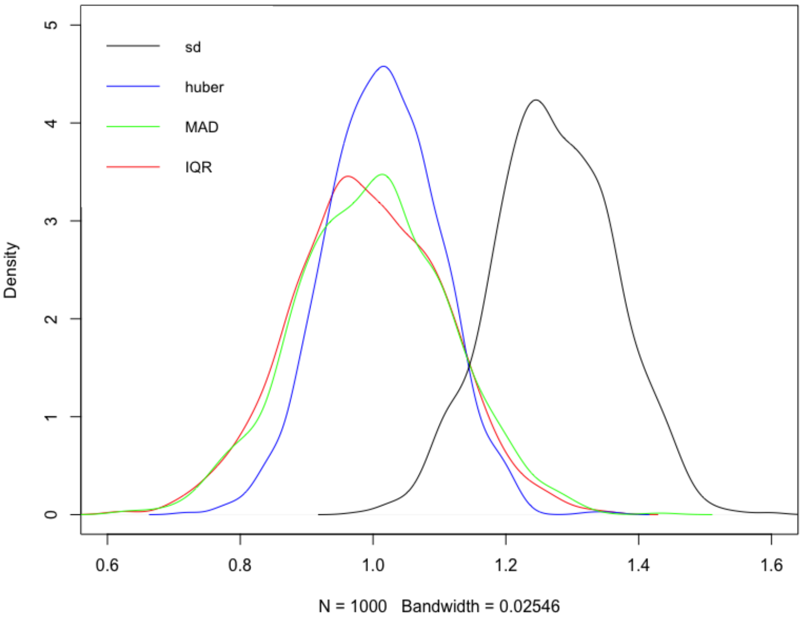


Figure 3.4: Sampling distributions of the standard deviation estimators for normally distributed data with one outlier

The results in figure 3.3 clearly show us how the robustness of the estimators influence their estimation. As we expected from the results from figure 3.1, the IQR and MAD estimator give a more fluctuated estimation than the sample standard deviation and Huber estimator until the outlier. After the outlier, the MAD, IQR and Huber estimations are closer to each other. The R-estimators become more accurate as they receive more data.
In figure 3.4 we can see that the sample standard deviation clearly explodes from one outlier and the estimation remains high after the outlier. On the other hand, the outlier filters from the other estimators do their work, as we see only a small reaction to the outlier and a better estimation of the standard deviation.

3.1.2 Logistic distribution

Another example from the location-scale family is the logistic distribution. Defining $\psi(x) = \dfrac{e^x}{(1+e^x)^2}$, then

$$f(x|\sigma) = \frac{1}{\sigma}\psi(\frac{x}{\sigma}) = \frac{e^{\frac{x}{\sigma}}}{\sigma(1+e^{\frac{x}{\sigma}})^2}$$

is the probability density function for the logistic distribution with location $\mu = 0$ and scale σ. This means that if the distribution of the single asset is logistic, the value at risk is defined as:

$$VaR_\alpha = P \cdot (\sigma\Psi^{-1}(\alpha)) = P \cdot (\sigma \ln(\frac{\alpha}{1-\alpha})).$$

We also ran a 1,000 simulations for the logistic distribution to determine the sampling distribution of the estimators. The simulations are with 100 observations from the logistic distribution with location, l=0, 0 and scale, s = 1. This means that the standard deviation is $\sigma = \dfrac{\pi s}{\sqrt{3}} = \dfrac{\pi 1}{\sqrt{3}} \approx 1.81$. In figure 3.5, the sampling distributions of the different estimators are shown.

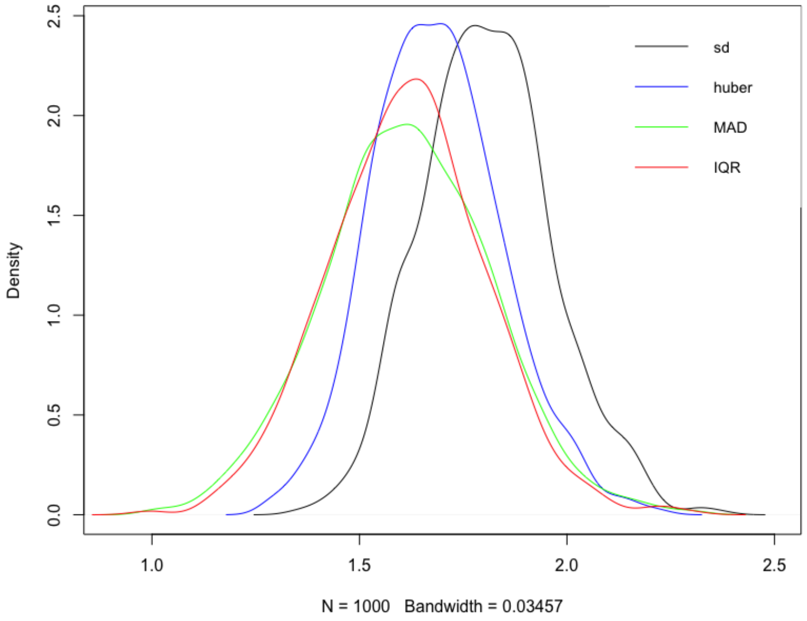


Figure 3.5: Sampling distributions of the standard deviation estimators for logistically distributed data

We can see that when the data follows the logistic distribution, the results are very similar to the results with the normal distribution. To see how they react to anomalies, we will simulate the behaviour for a polluted data set drawn from the logistic distribution. Each of the 1,000 simulations are with 100 observations from the logistic distribution with location 0 and scale 1, and data point 60 is replaced by 8 times the value of the sample standard deviation.

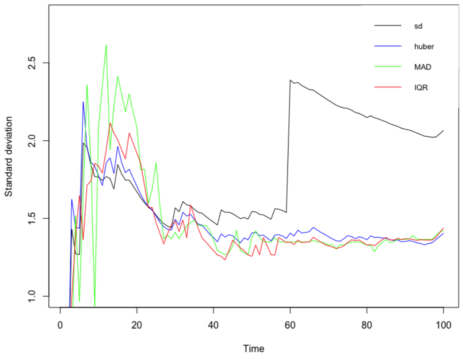


Figure 3.6: Example of the standard deviation estimation given a polluted normally distributed data set

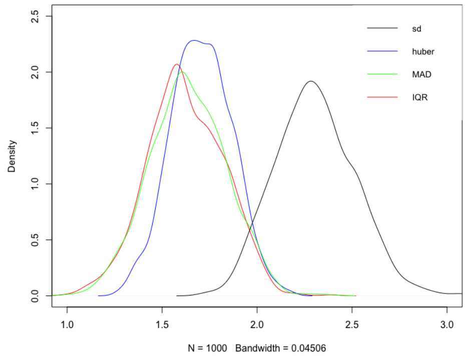


Figure 3.7: Sampling distributions of the standard deviation estimators for logistically distributed data with one outlier

The results from figures 3.6 and 3.7 are in line with our expectations from the previous figures. It shows that the Huber, IQR, and MAD estimators are more robust, they do not overreact to the outlier, while the sample standard deviation again explodes. The other similarity with the results for the normally distributed data set is that the IQR and MAD estimators have a wider distribution then the Huber estimator. Which means that the IQR and MAD will show more fluctuations in their estimation.

It is clear that the estimations of the IQR and MAD estimators are very close to each other. In the simulations to come, we will only show the MAD estimator. The reason for this is that the graphs of these two estimators are so close, it becomes hard to distinguish between the lines. The reason we choose the MAD estimator is that this estimator is also commonly used in the financial sector. In appendix A, the results with all four estimators are shown.

3.2 Portfolio of 1 asset

We will discuss three data sets and analyse the performance of the estimators. We will also briefly describe the historical moments that caused the extreme fluctuation of the price data itself. This will give insight in to why these extreme price changes occur.

The assumption that the mean of the data set equals zero, makes that the VaR is a constant times the standard deviation. This means it suffices to look exclusively at the standard deviation estimators. As we discussed, we will also look at different lookback periods. We will simulate the estimations with a 'no limit lookback' period and with the commonly used '100 day lookback' period.

3.2.1 Crude Oil

The first data set we will analyse contains the daily return rates of crude oil from 30 March 1983 until 12 August 2016. In the period from 30 March 1983 to 12 August 2016 a few events happened that caused extreme price changes. In figure 3.8 we will visualise the data and we have circled the following events:

- *17 Januari 1991*, marked by a yellow circle. Due to the successful raids of the American air force on Iraq, the oil prices decreased extremely. The oil market feared a disruption of the oil supply and damage to the oil installation in the middle east.[8]

- *6 June 2008*, marked by a red circle. On this day, another war caused the oil market to fear the disruption of the supply. [9]

- *29 September 2008*, marked by an orange circle. This date will probably be an outlier in many of our data sets since this was not an event related to oil, but it affected the global markets. The drop happened after the House of Representatives voted down a $700 billion bank bailout plan.[10]

- *5 May 2011*, marked by a blue circle. The oil prices dropped because of weak economic data. The investors feared that the American economy was slowing down, which in hindsight was correct. [11]

- *29 June 2012*, marked by a green circle. A big increase in the oil price occurred on this date. The reason is a deal by European leaders to help Euro zone banks. [12]

- *28 November 2014*, marked by a purple circle. On this day, OPEC ministers decided not to change the output ceiling for the oil market. The ceiling is at least 1 million above the estimates of demand, resulting in excess supply. The decision did not force oil producers to stop overproducing, so the prices dropped.[13]

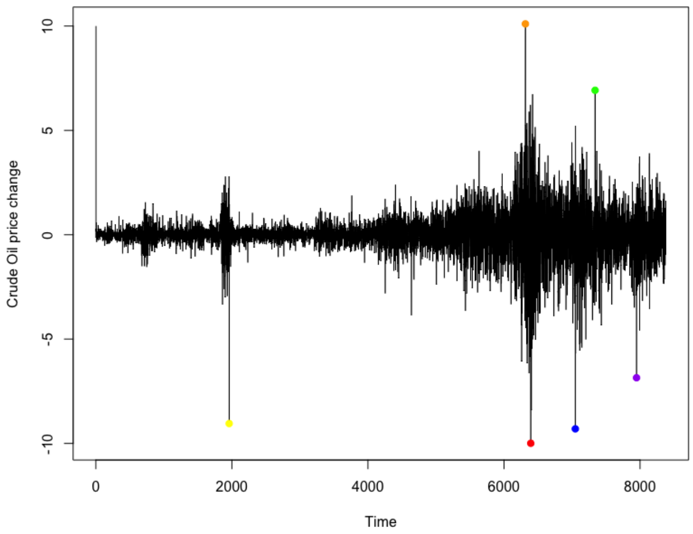


Figure 3.8: The crude oil price changes

The results from the estimators for 'no limit lookback' and '100 days lookback' are displayed in figure 3.9 below.

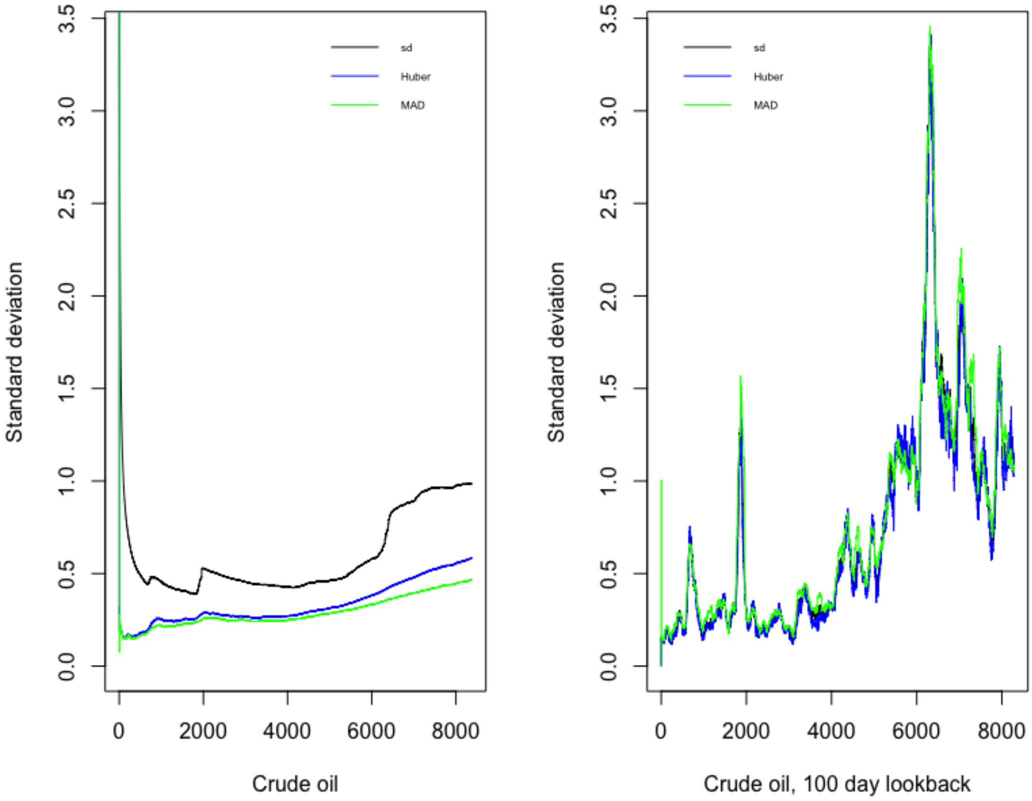


Figure 3.9: Standard deviation estimation for Crude Oil data set. Left: unconstrained lookback window. Right: 100 day lookback window

In figure 3.9 it is visible that the extreme price changes have an impact on the estimators, especially on the sample standard deviation. In both figures, we can see an increase of the estimations when the outliers occur. The price data with these extreme outliers clearly influence the behaviour of the estimators. If we only look at the '100 day lookback' period, it is harder to see the difference between the estimators, but if you look at the extreme values, there is a difference between the estimators.
What also stands out is the difference between the estimations given the two different lookback periods. When the estimators can use all data available, the estimation reacts less extreme to the outliers but the disadvantage to this can also mean that they are underestimating the standard deviation.

3.2.2 S&P Index

The Standard & Poor's 500 is an American stock market index that is based on the market capitalisation of 500 large companies that have common stock listed on the NYSE or NASDAQ. [14] Our data set contains daily returns from 3 January 1950 until 12 August 2016. In figure 3.10 we have accentuated the following three events:

- *19 October 1987*, marked by a blue circle. This day is called Black Monday, markets all around the world crashed. There is not a specific cause for this crash. Economists still debate what the main influence has been, but possible causes are program trading, overvaluation and market psychology. [15]

- *29 September 2008*, marked by a purple circle. This date also appeared in the analysis of crude oil. The House of Representatives voted down a $700 billion bank bailout plan.[10]

- *13 October 2008*, marked by a green circle. This date marks the biggest increase of the S&P in this data set. The cause is the opposite of what happened on 29 September that year. The European governments as well as the American government announced plans to help the banks with loans.[16]
The results from the simulation are displayed in figure 3.11 below.

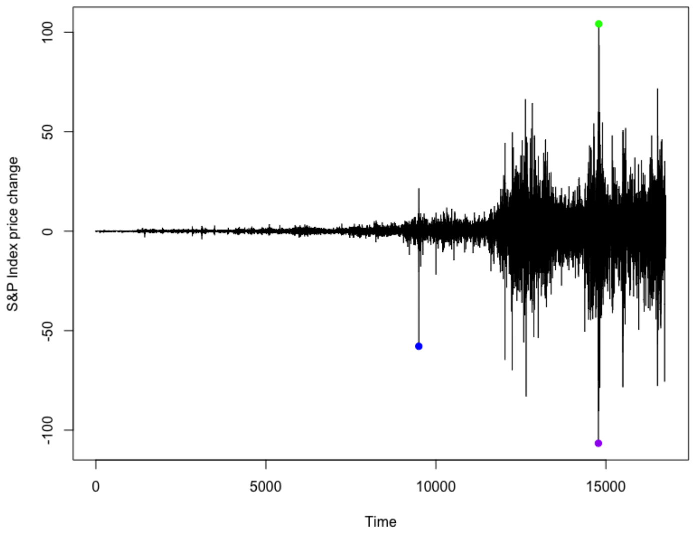


Figure 3.10: S&P Index price changes

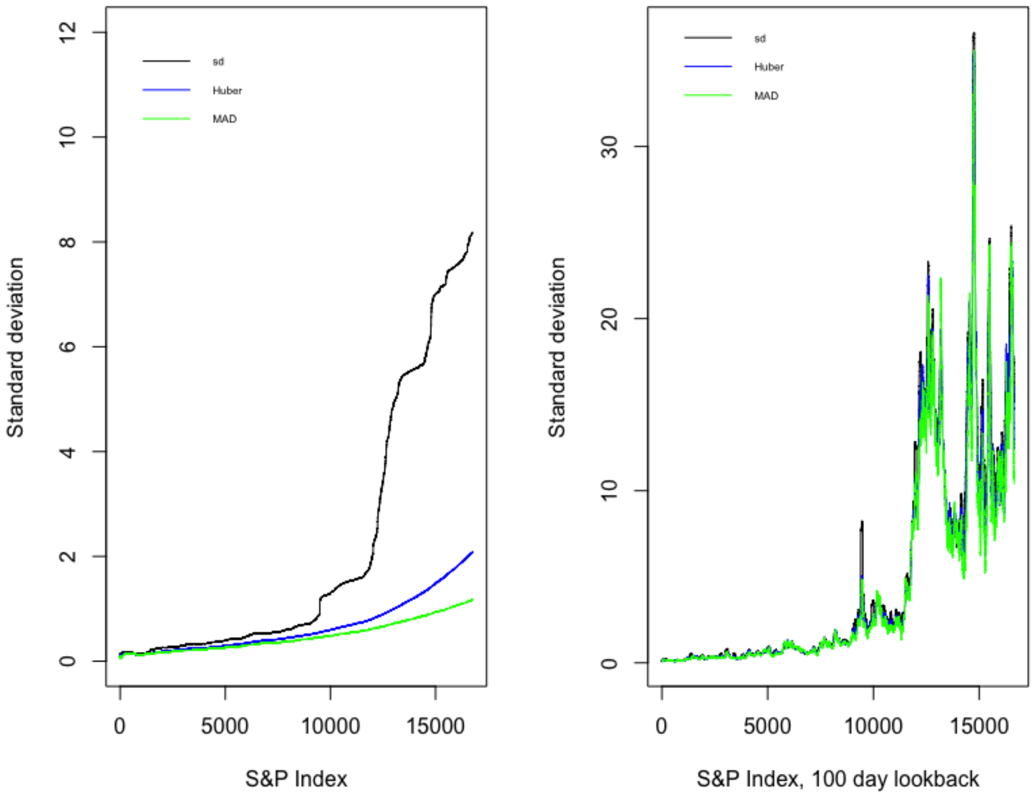


Figure 3.11: Standard deviation estimation for S&P data set. Left: unconstrained lookback window. Right: 100 day lookback window

The difference between the 'no limit lookback' and the '100 day lookback' is a lot bigger than for the previous data set. The reason the difference is so big is the change in the price changes in the second half of the data set. This change in behaviour makes that all the estimators with 'no limit lookback' greatly underestimate the standard deviation. The '100 day lookback' clearly has our preference as it is not affected by the historical data.

We can see that when we give the estimators the '100 day lookback' window, their estimations are very similar for the majority of the time. To show that around an outlier, the estimators give a really different estimation, we will zoom in on the graphs and look at the time frame in which the first extreme outlier occurs.

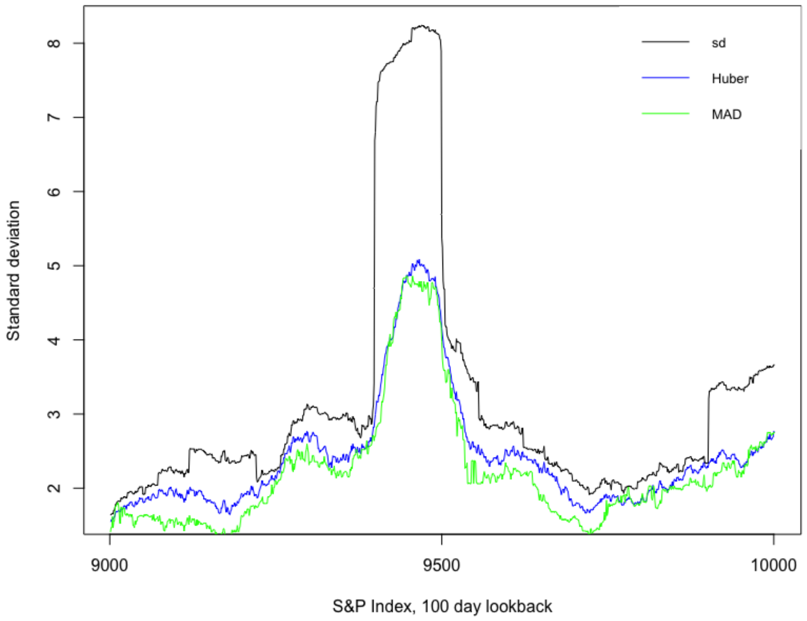


Figure 3.12: Standard deviation estimation for S&P data set, '100 day lookback' window

In figure 3.12 we can see that the outlier caused the sample standard deviation to explode and it was significantly higher than the MAD and Huber estimator for more than 100 days. We can also see that for most of the time, the MAD estimation is lower then the Huber estimation.

A data set that has such extreme outliers that it will show the difference between the estimators even more clearly is the price data from the Swiss Franc.

3.2.3 Swiss Franc

The last data set that we will look at contains the daily returns of the Swiss Franc from 2 January 2002 until 12 August 2016. The four events that we have marked in figure 3.13 are the following:

- *12 March 2009*, marked by a blue circle. On this date, the Swiss National Bank eased monetary policy. Its actions included a policy-rate cut, the purchase of Swiss private-sector bonds, and foreign-exchange interventions. Immediately after announcing the policy changes, the bank aggressively bought Euros in the foreign-exchange market. The Swiss Franc depreciated sharply. [21]

- *9 August 2011*, marked by a purple circle. While most other markets decreased in August 2011, the Swiss Franc increased. The reason is that this currency has the reputation of being a safe haven. The announcement of the Federal Reserve of the US to freeze US interest rates for the next 2 years led to an increased interest from foreign investors to buy Swiss Francs, which caused this extreme increase. [22]

- *6 September 2011*, marked by a green circle. In an attempt to protect the Swiss economy from the European debt crises, the Swiss National Bank devaluated the Franc, pledging to buy unlimited quantities of foreign currencies.[23]

- *15 January 2015*, marked by an orange circle. On this day, the Swiss National Bank announced that it would no longer hold the Swiss Franc at a fixed exchange rate with the Euro. This announcement was unexpected, causing a big collapse of the Swiss Franc as well as the Swiss stock market.[24]

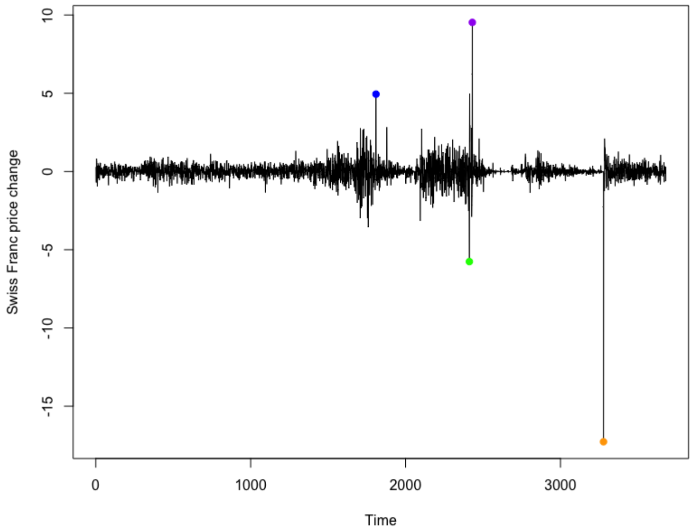


Figure 3.13: Swiss Franc price changes

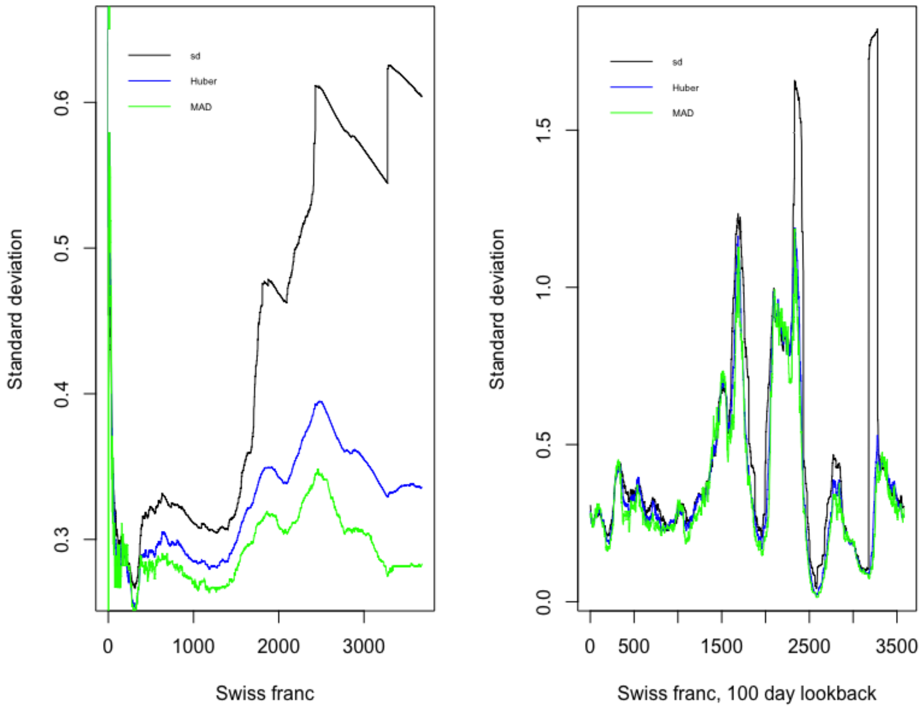


Figure 3.14: Standard deviation estimation for Swiss Franc data set. Left: unconstrained lookback window. Right: 100 day lookback window

The similarity between the results in figure 3.14 and the previous simulations is that the 'no limit lookback' estimations underestimate the standard deviation. The difference, on the other side, is that the outliers are so extreme that the difference between the reaction of the sample standard deviation and the other estimators is much bigger and clearly visible. This is an example of a situation in which the investor - if he would use the sample standard deviation estimator - would have to sell its shares due to the VaR limits. We can see that Value at Risk did not actually explode because the outlier was a single outlier.

Chapter 4

Portfolio of 2 assets

In this chapter we will consider the two dimensional problem of a portfolio of two assets. As said before, the Value at Risk in this situation is not only determined by the standard deviation of the assets but is also influenced by the correlation between the assets. Before we look at how the different estimators react to data drawn from the location-scale family, we will take a closer look at the variance of a portfolio of two assets. We will also look at a completely different way to model assets: the Factor model. Last, we will simulate the VaR estimation of a portfolio of assets with real price data.

4.1 Variance 2 assets

Assume we have two assets, X and Y, and we define $Z = [X, Y]^T$.
We define the covariance matrix of Z as: $\Sigma_Z = \begin{bmatrix} \Sigma_{X,X} & \Sigma_{X,Y} \\ \Sigma_{Y,X} & \Sigma_{Y,Y} \end{bmatrix}$, where $\Sigma_{X,X} = \sigma_X^2$, $\Sigma_{Y,Y} = \sigma_Y^2$ and $\Sigma_{X,Y} = \Sigma_{Y,X} = \text{cov}(X,Y) = \rho_{X,Y}\sigma_X\sigma_Y$.

We also introduce $w = [w_X, w_Y]^T$, where w_X is the weight of asset X and w_Y the weight of asset Y. Note that $w_X + w_Y = 1$.
In the situation $Z_w = w^T Z$, we can compute the variance of Z_w in the following way:

$$\Sigma_{Z_w} = w^T \Sigma_Z w.$$

$$\begin{aligned}
\text{Proof:} \qquad w^T\Sigma_Z w &= [w_X, w_Y] \begin{bmatrix} \Sigma_{X,X} & \Sigma_{X,Y} \\ \Sigma_{Y,X} & \Sigma_{Y,Y} \end{bmatrix} \begin{bmatrix} w_X \\ w_Y \end{bmatrix} \\
&= [w_X\Sigma_{X,X} + w_Y\Sigma_{Y,X}, w_X\Sigma_{X,Y} + w_Y\Sigma_{Y,Y}] \begin{bmatrix} w_X \\ w_Y \end{bmatrix} \\
&= w_X^2\Sigma_{X,X} + w_Y^2\Sigma_{Y,Y} + 2w_Xw_Y\Sigma_{X,Y} \\
&= Var(w_XX + w_YY) \\
&= \Sigma_{Z_w}
\end{aligned}$$

This shows that the weights of the assets influence the variance and thus the VaR of the portfolio. We will show that we can find weights such that it minimises the variance. Since $w_X + w_Y = 1$ we can write $\Sigma_{Z_w} = Var(w_XX + (1 - w_X)Y)$.

$$\min_w \;\; Var(wX + (1-w)Y)$$

$$Var(wX + (1-w)Y) = w^2\Sigma_{X,X} + (1-w)^2\Sigma_{Y,Y} + 2w(1-w)\Sigma_{X,Y}$$

$$= w^2(\Sigma_{X,X} - 2\Sigma_{X,Y} + \Sigma_{Y,Y}) + 2w(\Sigma_{X,Y} - \Sigma_{Y,Y}) + \Sigma_{Y,Y}$$

Minimising this quadratic function in terms of w gives us

$$f'(w) = 2 * w * (\Sigma_{X,X} - 2\Sigma_{X,Y} + \Sigma_{Y,Y}) + 2 * (\Sigma_{X,Y} - \Sigma_{Y,Y})$$

$$w_{min} = \frac{-(\Sigma_{X,Y} - \Sigma_{Y,Y})}{\Sigma_{X,X} - 2\Sigma_{X,Y} + \Sigma_{Y,Y}}$$

Note that we do not consider the possibility of going short, so the relationship between the weights of the assets and the variance can be seen in figure 4.1. For this figure we assumed that $\Sigma_{X,X} = 3, \Sigma_{Y,Y} = 2$.

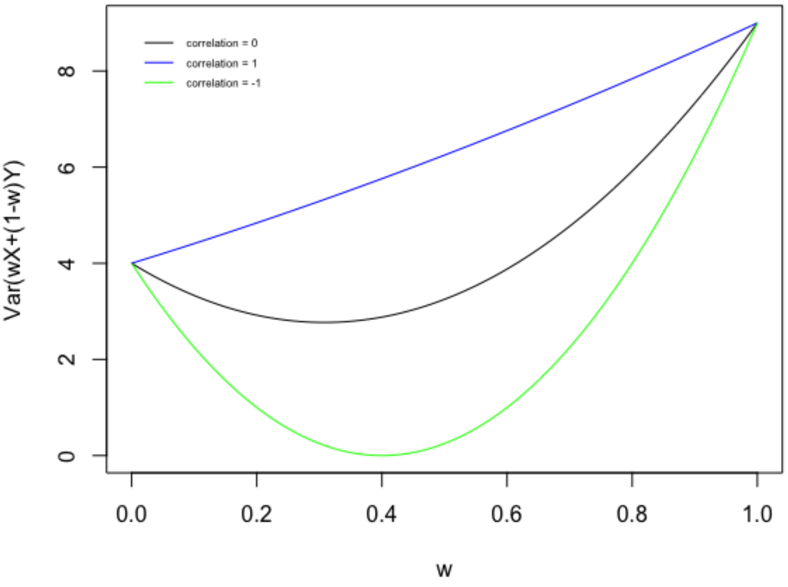


Figure 4.1: Relationship between the weights and the variance

It is clear that it is more complicated to estimate the value at risk in this situation. We need estimators $\hat{\sigma}_X, \hat{\sigma}_Y$ for the scale estimate, an estimator $\hat{\rho}_{X,Y}$ to estimate the correlation between X and Y and the composition of our portfolio has an influence on the variance.

4.2 Location-scale families

If our two assets follow any distribution from the location-scale family, we can determine the VaR very similar to the way we did for 1 asset.

We assume that both assets are from the same location-scale family. This means that given fixed random variables W and V, both with density function $\psi(x)$, we can define a location-scale family based on $\psi(x)$, location parameter $a = 0$ and scale parameter b. We define the random variables of the two assets as:

$$X = b_X V$$

$$Y = b_Y W$$

$$Z = [X, Y]^T, \ w = [w_X, w_Y]^T$$

We know that the quantile function of the assets are of the form: $Q(\alpha) = F^{-1}(\alpha) = b\Psi^{-1}(\alpha)$.

So we can determine the VaR of the portfolio as:

$$\begin{aligned} VaR_\alpha &= \sqrt{\Sigma_{Z_w}} \cdot \Psi^{-1}(\alpha), \text{ with} \\ \Sigma_{Z_w} &= w_X^2 b_X^2 \sigma_V^2 + w_Y^2 b_Y^2 \sigma_W^2 + 2\ w_X w_Y\ \rho_{X,Y}(b_X\sigma_V)(b_Y\sigma_W). \end{aligned}$$

4.2.1 Normal distribution

We first assume that the two assets in our portfolio are independent normal distributed.
$X \sim N(0, \sigma_X^2), Y \sim N(0, \sigma_Y^2)$, $Z = [X, Y]^T$ and the weights of assets X and Y are respectively w_X and w_Y. Then the portfolio $Z_w \sim N(0, w^T \Sigma_Z w)$.
The variance of the weighted combination of the assets is calculated as:

$$\Sigma_{Z_w} = w^T \Sigma_Z w = w_X^2 \sigma_X^2 + w_Y^2 \sigma_Y^2 + 2 w_X w_Y \rho_{X,Y} \sigma_X \sigma_Y.$$

We assume $w_X = w_Y = \frac{1}{2}$.
If we want to look at the different correlation estimators in VaR estimations, we have to look at them in combination with the standard deviation estimators. We have three different standard deviation estimators: the sample standard deviation, the Huber estimator and the MAD estimator. And we will look at three different correlation estimators: the Pearson correlation, the Kendall correlation and the three options within the covRob function. In figure B.1 we can see the Σ_{Z_w} estimations for the 9 different combinations. The results are based on 1,000 simulations of 100 observations from the multivariate normal distribution with mean 0 and covariance matrix $\Sigma = \begin{bmatrix} 4 & 2 \\ 2 & 3 \end{bmatrix}$. This means that the correlation is $\rho(X, Y) = \dfrac{\Sigma_{X,Y}}{\sigma_X \sigma_Y} = \dfrac{2}{\sqrt{3}\sqrt{4}} \approx 0.58$.

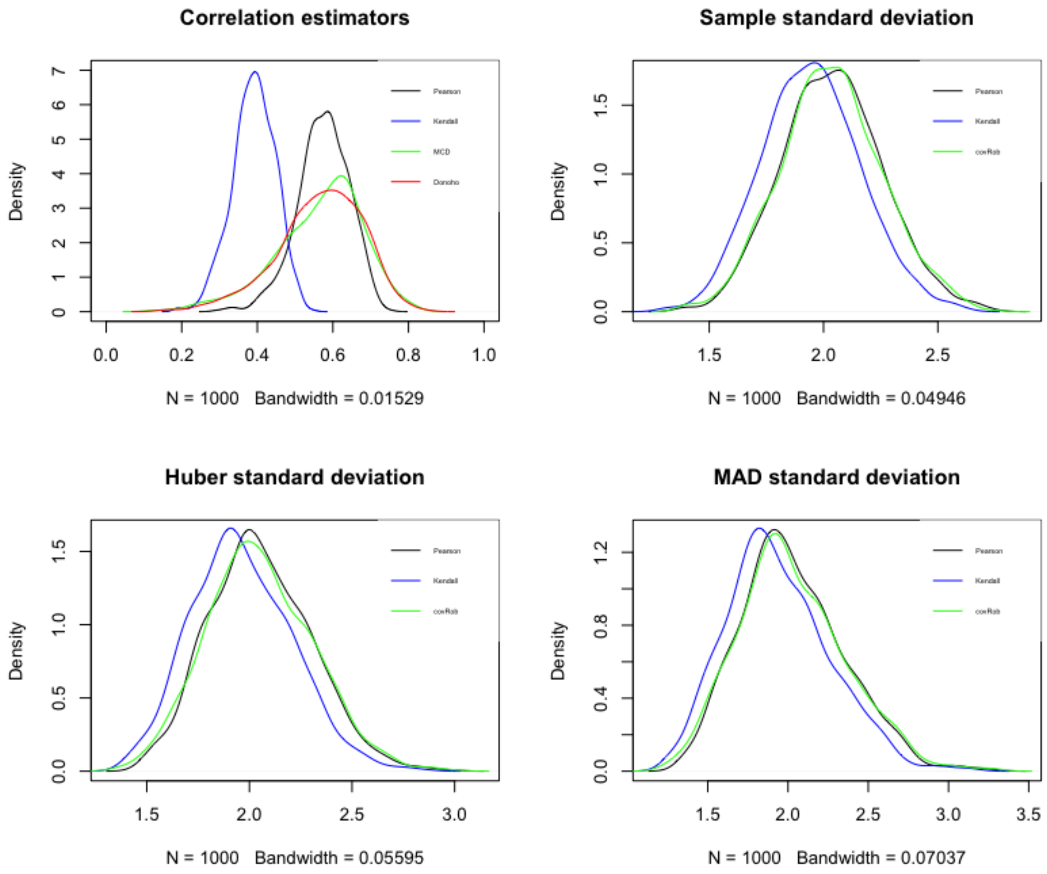


Figure 4.2: Results simulation multivariate normally distributed data.
Top left: Sampling distributions for the correlation estimators. Top right: Sampling distributions for Σ_{Z_w} with sample standard deviation and the different correlation estimators. Bottom left: Sampling distributions for Σ_{Z_w} with Huber estimator and the different correlation estimators. Bottom right: Sampling distributions for Σ_{Z_w} with MAD estimator and the different correlation estimators.

For completeness we show the sampling distribution for the two options of the covRob function in the top left graph, but we will let the covRob choose the best suited option for the Σ_{Z_w} calculations. In this figure we see that the distribution of the different correlation estimators are very different. We see that the Pearson and covRob estimators, on average, give a higher correlation than the Kendall estimator. When looking at the Σ_{Z_w} simulations, we see that the shape is influenced by the standard deviation. The correlation estimator can shift the distribution to the left or right.

We ran a 1,000 simulations in which we constructed polluted data sets to see how the estimators respond to an outlier. Each simulation is with 100 observations from the multivariate normal distribution with mean 0 and covariance matrix $\Sigma = \begin{bmatrix} 4 & 2 \\ 2 & 3 \end{bmatrix}$ The outlier added at $t = 60$ is 8 times the sample standard deviation. An example of one data set from the simulation is displayed in figure 4.3, the outlier is marked with a solid red circle. The results of the simulation are displayed in figure B.2.

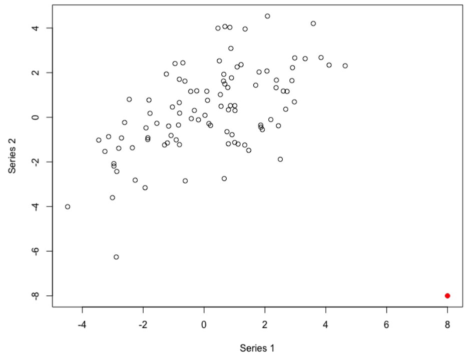


Figure 4.3: Example of a polluted normally distributed data used in the simulation

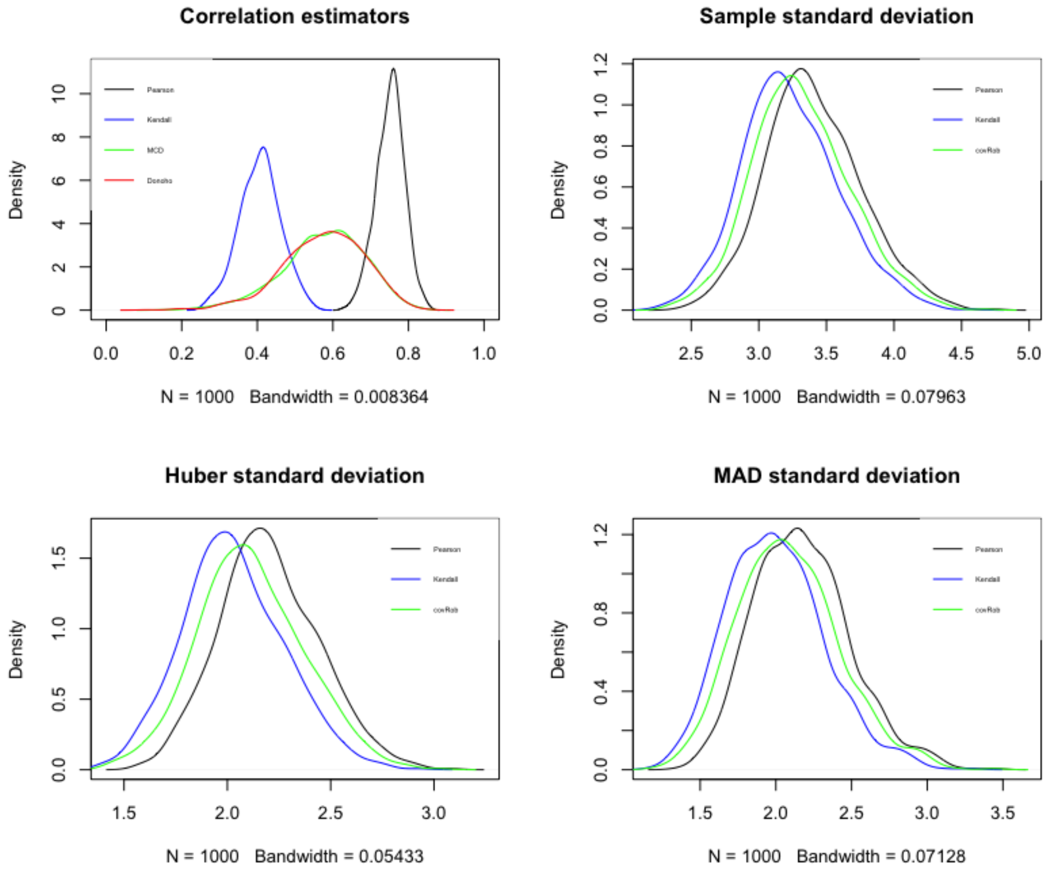


Figure 4.4: Results simulation multivariate normally distributed data with an outlier.
Top left: Sampling distributions for the correlation estimators. Top right: Sampling distributions for Σ_{Z_w} with sample standard deviation and the different correlation estimators. Bottom left: Sampling distributions for Σ_{Z_w} with Huber estimator and the different correlation estimators. Bottom right: Sampling distributions for Σ_{Z_w} with MAD estimator and the different correlation estimators.

If we look at the top left graph of figure B.2, we see that the distribution of the Pearson estimator changed. It overestimates the correlation and we can now clearly see a difference between the Pearson and covRob correlation estimators in the sampling distributions for Σ_{Z_w}.

The results for the logistic distribution are very similar and can be found in the appendix. For data sets from the location-scale family, the covRob function gives the most robust estimation for the correlation. The Kendall estimator is also resistant to outliers but it gives a lower correlation then the real value. The Pearson correlation works really well for samples without outliers but reacts heavily to a single outlier.

4.3 Factor model

Before we look at simulation with real data for this chapter, we will address a completely different way to model the assets in a portfolio. The Factor model is a more realistic but complex way to model the behaviour of an asset.
We assume that the price changes of the asset are a reaction to certain factors. This section is based on chapter 3 of [Quantitative Risk Management, 2005] [2]

Given factors $F_j, j = 1, .., M$, the factor model is defined as:

$$X_{k,t} = \sum_{j=1}^{M} \lambda_{k,j} F_{j,t} + \epsilon_{k,t}, \quad k = 1, .., N,$$

$$F_{j,t}, \epsilon_{k,t} \sim N(0,1), \quad j = 1, ..., M, \quad k = 1, ..., N$$

We assume that $M = N = 2$, which means the two assets depend on two factors and can be modelled as:

$$X_{i,t} = \lambda_{i,1} F_{1,t} + \lambda_{i,2} F_{2,t} + \epsilon_{i,t}, \quad i = 1, 2$$

With the property: $Var(X_{i,t}) = 1$. From this we can conclude what the Var($\epsilon_{i,t}$) is:

$$\begin{aligned} Var(X_{i,t}) &= Var([\lambda_{i,1}, \lambda_{i,2}] \begin{bmatrix} F_{1,t} \\ F_{2,t} \end{bmatrix} + \epsilon_{i,t}) \\ &= Var(\lambda_{i,1} F_{1,t}) + Var(\lambda_{i,2} F_{2,t}) + Var(\epsilon_{i,t}) \\ &= \lambda_{i,1}^2 Var(F_{1,t}) + \lambda_{i,2}^2 Var(F_{1,t}) + Var(\epsilon_{i,t}) \\ &= \lambda_{i,1}^2 + \lambda_{i,2}^2 + Var(\epsilon_{i,t}) \\ &= 1 \end{aligned}$$

From the last two lines we can conclude:

$$\text{Var}(\epsilon_{i,t}) = 1 - (\lambda_{i,1}^2 + \lambda_{i,2}^2)$$

It is clear that the variance of a single asset is controlled by $\lambda_{i,1}$ and $\lambda_{i,2}$. When we look at the whole model with two assets, the variance is still controlled by the dependencies of the assets to these factors.
We define $Z = [X_1, X_2]$, then the variance can be written as:

$$\text{Var}(Z) = \Lambda\Lambda^T + \text{Var}(\epsilon) = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$$

With $\text{Cov}(X_1, X_2) = \lambda_{1,1}\lambda_{2,1} + \lambda_{1,2}\lambda_{2,2} = \rho$.

To see how this model reacts to an outlier, we will run 1,000 simulations with and without the outlier and fit it to the Factor model. The simulation of the price data $X_{1,t}$ and $X_{2,t}$ are each with 100 observations with covariance matrix $\Sigma = \begin{bmatrix} 1 & 0.2 \\ 0.2 & 1 \end{bmatrix}$. The polluted data sets are constructed as in all previous simulations.

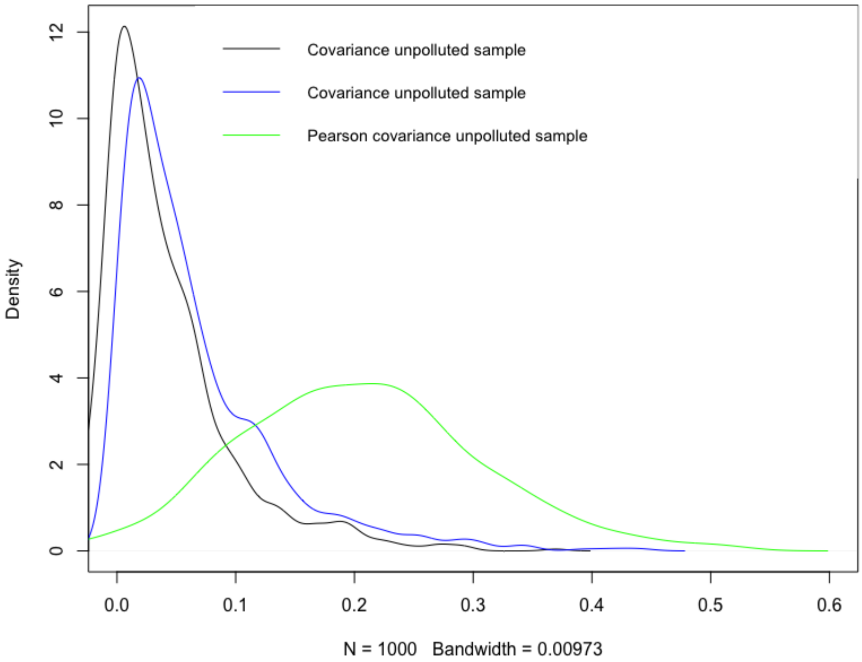


Figure 4.5: Sampling distributions of sample covariance estimation of the Factor model

We added the Pearson covariance estimation for comparison. It is clear that the factor model estimates a lower correlation then the Pearson correlation. A possible reason for this is that the fitted Factor models do not fit the sample very well. The information about the errors, $\epsilon_{1,t}$ and $\epsilon_{2,t}$, for the simulations are displayed in table 4.1.

	Minimum	Average	Maximum
$\epsilon_{1,t}$	0.1878	0.2043	0.2185
$\epsilon_{2,t}$	0.1494	0.1599	0.1703

Table 4.1: Standard error of the fitted factor model in simulation

From figures 4.5, we can see how this model reacts to an outlier. The overall covariance estimation of the Factor model increases and there are more extreme values.

4.4 Portfolio of 2 assets

We will simulate the VaR estimation using the data from the S&P500 Energy and the S&P500 Health care. We will show the results in the several graphs because of all the different estimators we can use. In chapter 3 we have seen that the 'no limit lookback' period does not give accurate VaR estimations so we will now only look at estimations with the '100 day lookback' period. We assume that the weights are equal.
Figures 4.6 and 4.7 display the data sets S&P500 Energy and the S&P500 Health care in which

we can see that they both have some outliers and both have a high variance. In figure 4.8 we can see that the data sets look positively correlated.

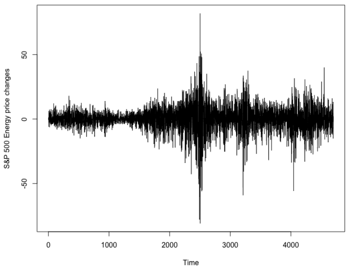


Figure 4.6

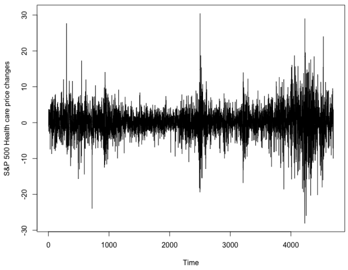


Figure 4.7

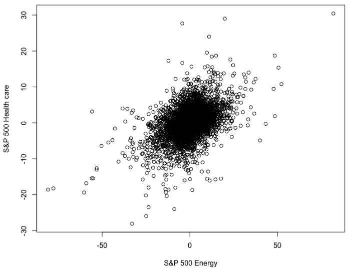


Figure 4.8

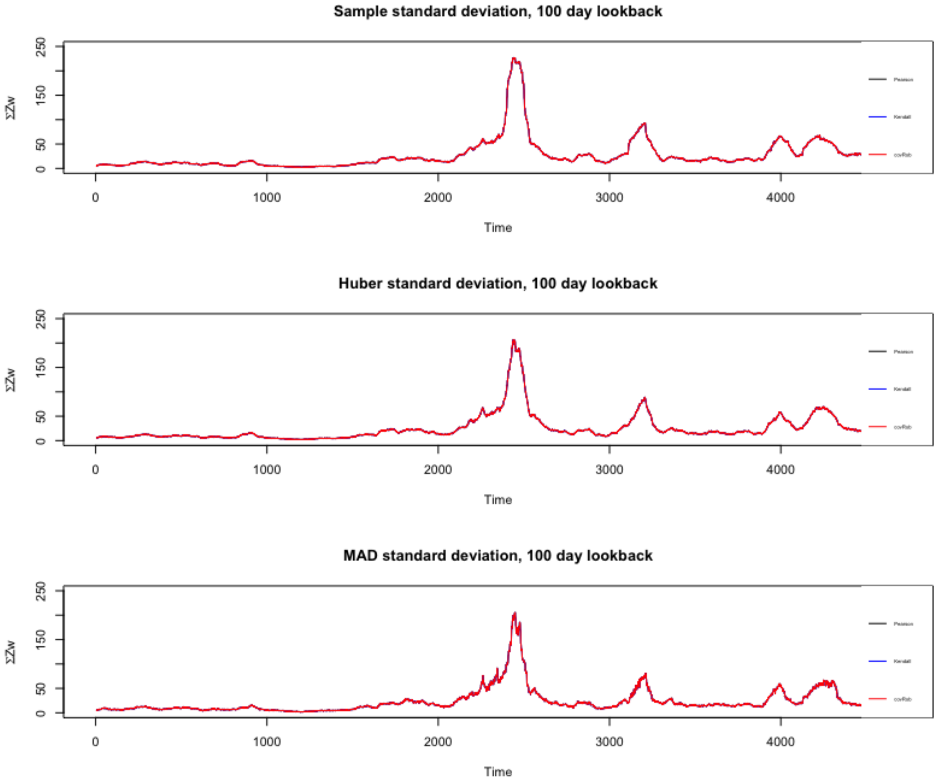


Figure 4.9: Σ_{Z_w} estimations for the S&P500 Energy and the S&P500 Health care data set

We can see that the outliers of the two data sets have an influence on the standard deviation estimators. They all show an increase around the data points that are clearly outliers in the data sets. The difference between the correlation estimators is hard to read from these graphs. The reason we cannot clearly see the difference is that the correlation is very small compared to the standard deviation. The correlation estimations are between -0,392 and 0,934, and the standard deviation estimates vary from 1,031 to 29,150. This means that the small differences between the correlation estimators hardly have an effect on the Σ_{Z_w}. To show that the correlation estimators hardly influence the Σ_{Z_w}, the difference between the Σ_{Z_w} estimations, given a fixed standard devation estimator, are displayed in table 4.2.

	Pearson - Kendall	Pearson - CovRob	Kendall - CovRob
Maximum absolute difference	0.20410	0.33370	0.23900
Average absolute difference	0.07989	0.04195	0.07892

Table 4.2: Difference between the Σ_{Z_w} estimations with a fixed standard deviation estimator

In situations like this, the standard deviation estimation is more important to the VaR calculation.

Chapter 5

Conclusion

We have shown in this report that different estimators react differently to extreme values. In the financial world, these differences have implications to the calculation of the Value at Risk. The VaR is the risk measure financial institutions like the European Central Bank use to set limits on the risk of portfolios. We have shown that if outliers occur, the robust estimators can filter these outliers, which results in a more accurate estimation. In practice, this means that if robust estimators are used, the VaR limits are not unnecessarily exceeded.

The sample standard deviation, MAD, IQR and Huber estimator are the standard deviation estimators that we have analysed. The sample standard deviation does not filter outliers, which leads to extreme reactions when anomalies in the data occurs. The MAD, IQR and Huber estimators, on the other hand are more resistant to outliers. They all filter outliers in a different way, leading to different outcomes. The results for data drawn from location-scale families supported this. We estimated the standard deviation for three real price data sets. These results also showed that the Huber or MAD estimator give a better estimation than the sample standard deviation.

To estimate the correlation between two assets, we have analysed the Pearson, Kendall, MCD and Donoho-Stahel estimators. The only correlation estimator without outlier filter is the Pearson estimator. In simulation with data drawn from a location-scale family we have seen that this estimator heavily reacts to an outlier. The results from simulation with real price data showed that for that example, the correlation estimator did not have a big influence on the VaR estimation. Using different standard deviation estimators, on the other hand, did influence the VaR estimation.

Bibliography

[1] Huber, P.J. Ronchetti, E.M. (2009) *Robust Statistics* (2nd Edition). Wiley series in probability and statistics

[2] McNeil, A.J. Frey, R Embrechts, P (2005) *Quantitative Risk Management.* Princeton University Press

[3] Rice, J.A. (2007) *Mathematical Statistics and Data Analysis* (3rd Edition). Brooks/Cole

[4] https://cran.r-project.org/web/packages/robust/robust.pdf

[5] Fox, J. (2016) *Applied regression analysis & Generalized linear models.* (3rd Edition) SAGE Publications

[6] Ricardo A. Maronna, and Victor J. Yohai (2015) *Robust and efficient estimation of multivariate scatter and location* National University of La Plata, La Plata, Argentina, University of Buenos Aires and CONICET, Argentina

[7] https://www.ecb.europa.eu/pub/pdf/scpops/ecbop174.en.pdf?ddf35b68fa3f29b87f99583c2157515b

[8] http://www.delpher.nl/nl/kranten/view?coll=ddd&identifier=ddd:010963661:mpeg21:a0001

[9] https://www.genealogieonline.nl/over-de-dag/2008/6/6

[10] http://money.cnn.com/2008/09/29/markets/markets_newyork/

[11] http://money.cnn.com/2011/05/05/markets/oil_prices/

[12] http://www.reuters.com/article/us-markets-oil-idUSBRE83H17O20120629

[13] http://www.reuters.com/article/us-opec-meeting-idUSKCN0JA0O320141128

[14] https://en.wikipedia.org/wiki/S%26P_500_Index

[15] https://en.wikipedia.org/wiki/Black_Monday_(1987)

[16] http://www.volkskrant.nl/economie/grootste-stijging-dow-jones-ooit a892345/

[17] http://seekingalpha.com/article/194587-golds-dramatic-rise-and-fall-in-1980s-why-its-important

[18] http://www.bloomberg.com/news/articles/2013-04-15/gold-extends-bear-market-losses-as-investors-reduce-etp-holdings

[19] https://www.buzzfeed.com/simonneville/the-last-time-the-pound-was-this-weak-was-1985-heres-what-ha?utm_term=.nwWgD0Men#.jbjZ9X0GE

[20] https://www.theguardian.com/business/1992/sep/17/emu.theeuro

[21] Michael D. Bordo, Owen F. Humpage, Anna J Schwarz (2015) *Strained Relations: US Foreign-Exchange Operations and Monetary Policy in the twentieth century.* University of Chicago Press

[22] https://en.wikipedia.org/wiki/August_2011_stock_markets_fall

[23] https://www.theguardian.com/business/2011/sep/06/switzerland-pegs-swiss-franc-euro

[24] http://www.economist.com/blogs/economist-explains/2015/01/economist-explains-13

Appendix A

Portfolio of 1 asset simulations

The results for the Crude Oil data set:

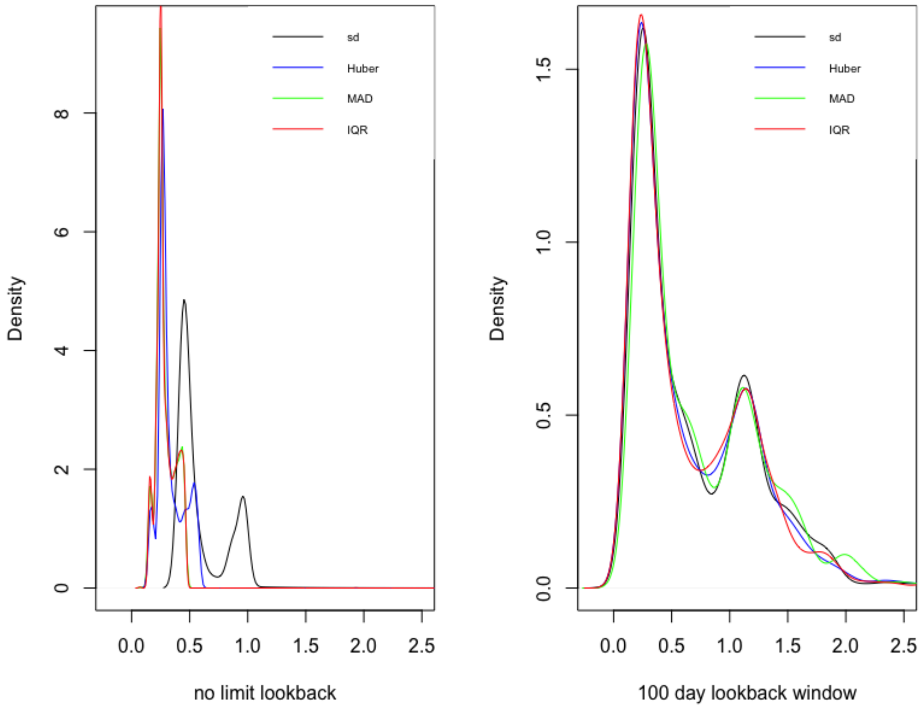


Figure A.1: Sampling distributions of the standard deviation estimators for Crude Oil data set. Left: unconstrained lookback window. Right: 100 day lookback window

The results for the S&P data set:

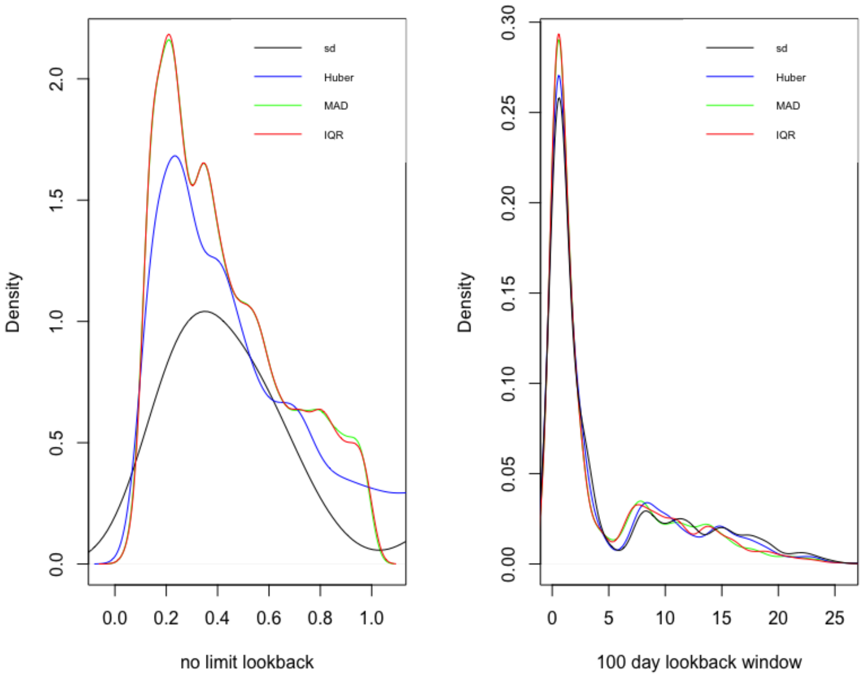


Figure A.2: Sampling distributions of the standard deviation estimators for S&P data set. Left: unconstrained lookback window. Right: 100 day lookback window

The results for the Swiss Franc data set:

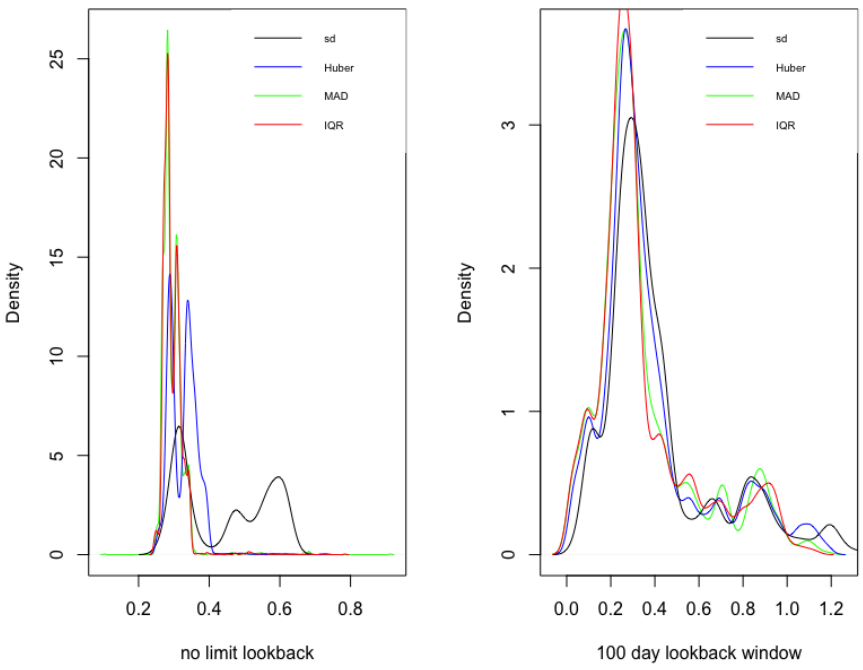
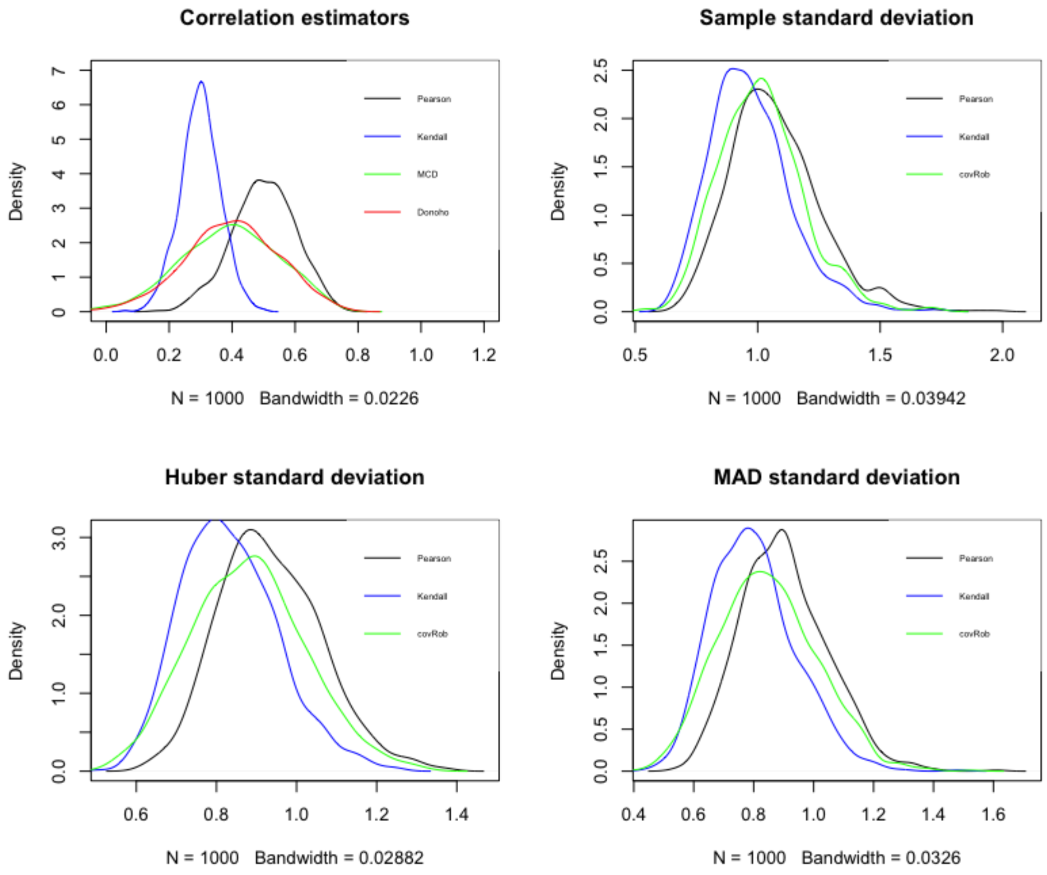


Figure A.3: Sampling distributions of the standard deviation estimators for Swiss Franc data set. Left: unconstrained lookback window. Right: 100 day lookback window

Appendix B

Portfolio 2 assets logistic distribution



Correlation estimators

Sample standard deviation

Huber standard deviation

MAD standard deviation

Figure B.1: Results simulation multivariate logistically distributed data.
Top left: Sampling distributions for the correlation estimators. Top right: Sampling distributions for Σ_{Z_w} with sample standard deviation and the different correlation estimators. Bottom left: Sampling distributions for Σ_{Z_w} with Huber estimator and the different correlation estimators. Bottom right: Sampling distributions for Σ_{Z_w} with MAD estimator and the different correlation estimators.

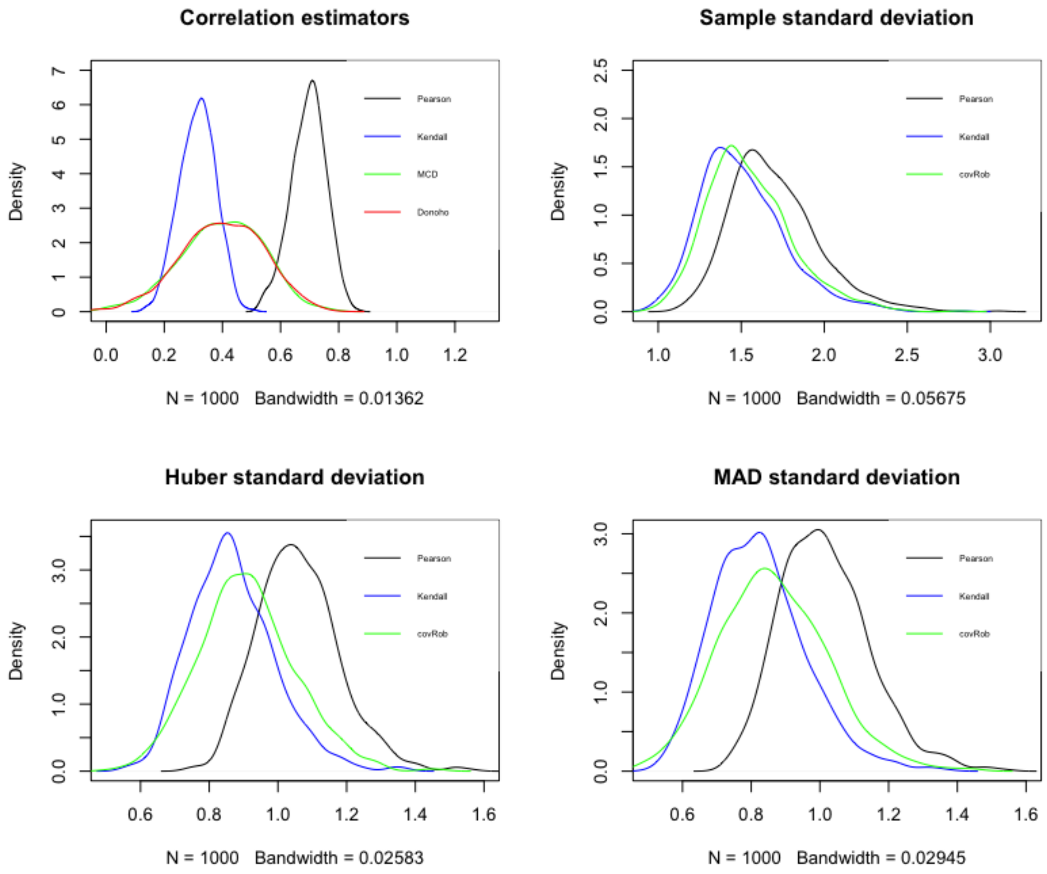


Figure B.2: Results simulation multivariate logistically distributed data with an outlier. Top left: Sampling distributions for the correlation estimators. Top right: Sampling distributions for Σ_{Z_w} with sample standard deviation and the different correlation estimators. Bottom left: Sampling distributions for Σ_{Z_w} with Huber estimator and the different correlation estimators. Bottom right: Sampling distributions for Σ_{Z_w} with MAD estimator and the different correlation estimators.

Appendix C

R Code

C.1 Simulations Portfolio 1 asset

C.1.1 Location scale family

```
############################
#    Normal Distribution    #
############################
sample_sd = rep(NA, 1000)
sample_sdM = rep(NA, 1000)
sample_sdL = rep(NA, 1000)
sample_sdIQR = rep(NA, 1000)
for(i in 1:1000){
  sample_sd[i] = sd(rnorm(100,mean=0,sd=1))
  sample_sdM[i] = hubers(rnorm(100,0,1))$s
  sample_sdL[i] = mad(rnorm(100,0,1))
  sample_sdIQR[i] = (IQR(rnorm(100,0,1))/1.349)
}
plot(density(sample_sdIQR),col="red",ylim=c(0,5.5), type = "l", main="")
lines(density(sample_sdM), col="blue")
lines(density(sample_sdL), col="green")
lines(density(sample_sd), col="black")
legend("topleft", legend=c("sd", "huber","MAD", "IQR"),col=c("black", "blue","gr


########################
#    Poluted sample    #
########################
sample_nsd = rep(NA, 1000) ; sample_nM = rep(NA, 1000)
sample_nL = rep(NA, 1000) ; sample_nIQR = rep(NA, 1000)
for(i in 1:1000){
  setn = rnorm(n=100, 0,1)
  setn[60]<-c(8*sd(setn))
  sample_nsd[i] = sd(setn)
  sample_nM[i] = hubers(setn)$s
  sample_nL[i] = mad(setn)
  sample_nIQR[i] = (IQR(setn)/1.349)
}
plot(density(sample_nIQR),col="red",ylim=c(0,5),xlim=c(0.6,1.6), type = "l", mai
lines(density(sample_nM), col="blue")
lines(density(sample_nL), col="green")
lines(density(sample_nsd), col="black")
legend("topleft", legend=c("sd", "huber","MAD", "IQR"),col=c("black", "blue","gr


############################
#   Logistic distribution  #
############################
sampleL_sd = rep(NA, 1000)
sampleL_sdM = rep(NA, 1000)
sampleL_sdL = rep(NA, 1000)
sampleL_sdIQR = rep(NA, 1000)
for(i in 1:1000){
  sampleL_sd[i] = sd(rlogis(100,location=0,scale=1))
  sampleL_sdM[i] = hubers(rlogis(100,location=0,scale=1))$s
  sampleL_sdL[i] = mad(rlogis(100,location=0,scale=1))
  sampleL_sdIQR[i] = (IQR(rlogis(100,location=0,scale=1))/1.349)
}

#######################
#   Poluted sample    #
#######################
sample_lsd = rep(NA, 1000) ; sample_lM = rep(NA, 1000)
sample_lL = rep(NA, 1000) ; sample_lIQR = rep(NA, 1000)
for(i in 1:1000){
   setl = rlogis(n=100, 0,1)
   setl[60]<-c(8*sd(setl))
  sample_lsd[i] = sd(setl)
  sample_lM[i] = hubers(setl)$s
  sample_lL[i] = mad(setl)
  sample_lIQR[i] = (IQR(setl)/1.349)
}
```

C.1.2 Price data

```
#######################
#      CRUDEOIL       #
#######################
crudeoil = read.csv("~/Desktop/crudeoil.csv",1)
oil =c(10);p=1
par(mfrow=c(1,1))
plot(density(oil), main="", xlab="")
plot.ts(oil, type= "l",ylab="Crude␣Oil␣price␣change")
points(1960,oil[1960],col = "yellow",pch=19)
points(6315,oil[6315],col = "orange",pch=19)
points(6394,oil[6394],col = "red",pch=19)
points(7049,oil[7049],col = "blue",pch=19)
points(7340,oil[7340],col = "green",pch=19)
points(7949,oil[7949],col = "purple",pch=19)
while(length(oil)<8379){
   oil = rbind(oil, c(crudeoil$CODEC[(p+1)]-crudeoil$CODEC[p]))
```

```
  p=p+1
}

#PLOT ESTIMATORS, FULL RANGE AND 100 DAYS
sl = c(0); slM = c(0); slL = c(0);slR = c(0); setextra = numeric(0);
p=101
while(p<8379){
  setextra <- oil[(p-100):p];
  sl = rbind(sl, c(sd(setextra)));
  slM = rbind(slM, c(hubers(setextra, k=1.5)$s));
  slL = rbind(slL, c(mad(setextra)));
  slR = rbind(slR, c(IQR(setextra)/1.349))
  p=p+1;
}
#ALL DATA
sl2 = numeric(0); slM2 = numeric(0); slL2 = numeric(0);slR2 = numeric(0); setext
p=2
while(p<8379){
  setextra2 <- oil[1:p];
  sl2 = rbind(sl2, c(sd(setextra2)));
  slM2 = rbind(slM2, c(hubers(setextra2, k=1.5)$s));
  slL2 = rbind(slL2, c(mad(setextra2)));
  slR2 = rbind(slR2, c(IQR(setextra2)/1.349))
  p=p+1;
}
par(mfrow=c(1,2))
plot.ts(sl2, type = "l", ylim=c(0, 3.4),xlab="Crude oil",ylab="Standard deviatio
lines(slM2, col="blue")
lines(slL2, col="green")
legend(4500, 3.5, legend=c("sd", "Huber","MAD"),col=c("black","blue", "green"),

plot.ts(slM, col="black",ylim=c(0, 3.4),xlab="Crude oil, 100 day lookback",ylab=
lines(slL, col="blue")
lines(sl, col="green")
legend(0, 3.5, legend=c("sd", "Huber","MAD"),col=c("black","blue", "green"), lty


#######################
#        SPINDEX      #
#######################
spindex1 = read.csv("~/Desktop/spindex.csv",1)
p=1;spindex=numeric(0);
while(length(spindex)<16762){
  spindex = rbind(spindex, c(spindex1$PRIJS[(p+1)]-spindex1$PRIJS[p]))
  p=p+1
}
par(mfrow=c(1,1))
plot(density(spindex), main="", xlab="")
plot.ts(spindex,ylab="S&P Index price change")
points(9498,spindex[9498],col = "blue",pch=19)
points(14790,spindex[14790],col = "green",pch=19)
points(14780,spindex[14780],col = "purple",pch=19)

#PLOT ESTIMATORS, FULL RANGE AND 100 DAYS
sp = numeric(0); spM = numeric(0); spL = numeric(0);spR = numeric(0); setextra =
p=2
while(length(setextra)<16761){
  setextra <- spindex[1:p];
  sp = rbind(sp, c(sd(setextra)));
  spM = rbind(spM, c(hubers(setextra, k=1.5)$s));
  spL = rbind(spL, c(mad(setextra)));
  spR = rbind(spR, c(IQR(setextra)/1.349))
  p=p+1;
}
sp1 = numeric(0); spM1 = numeric(0); spL1 = numeric(0);spR1 = numeric(0); setext
p=101
while(p<16761){
  setextra1 <- spindex[(p-100):p];
  sp1 = rbind(sp1, c(sd(setextra1)));
  spM1 = rbind(spM1, c(hubers(setextra1, k=1.5)$s));
  spL1 = rbind(spL1, c(mad(setextra1)));
  spR1 = rbind(spR1, c(IQR(setextra1)/1.349))
  p=p+1;
}

####################################
#        OUTLIER BEHAVIOUR         #
####################################

par(mfrow=c(1,1))
plot(sp1[9000:10000], xaxt="n" ,type="l", col="black",xlab="S&P␣Index,␣100␣day␣l
axis(1, at = seq(0,1000, by =500), labels=seq(9000, 10000, by = 500))
lines(spM1[9000:10000], col="blue")
lines(spL1[9000:10000], col="green")
legend("topright", legend=c("sd", "Huber␣","MAD"),col=c("black", "blue", "green"


######################
#      SWISS         #
######################

swiss1 = read.csv("~/Desktop/swissfranc.csv",1)
p=1;swiss=numeric(0);
while(length(swiss)<3680){
  swiss = rbind(swiss, c(swiss1$CODEC[(p+1)]-swiss1$CODEC[p]))
  p=p+1
}
par(mfrow=c(1,1))
plot(density(swiss), main="", xlab="")
plot.ts(swiss,ylab="Swiss␣Franc␣price␣change")
points(1809,swiss[1809],col = "blue",pch=19)
points(2412,swiss[2412],col = "green",pch=19)
points(2431,swiss[2431],col = "purple",pch=19)
points(3278,swiss[3278],col = "orange",pch=19)


#PLOT ESTIMATORS, FULL RANGE
sp = numeric(0); spM = numeric(0); spL = numeric(0);spR = numeric(0); setextra =
p=2
while(length(setextra)<3680){
  setextra <- swiss[1:p];
  sp = rbind(sp, c(sd(setextra)));
  spM = rbind(spM, c(hubers(setextra, k=1.5)$s));
  spL = rbind(spL, c(mad(setextra)));
  spR = rbind(spR, c(IQR(setextra)/1.349))
  p=p+1;
}
#PLOT ESTIMATORS, 100 DAY LOOKBACK
sp1 = numeric(0); spM1 = numeric(0); spL1 = numeric(0);spR1 = numeric(0); setext
p=101
while(p<3680){
  setextra1 <- swiss[(p-100):p];
  sp1 = rbind(sp1, c(sd(setextra1)));
  spM1 = rbind(spM1, c(hubers(setextra1, k=1.5)$s));
  spL1 = rbind(spL1, c(mad(setextra1)));
  spR1 = rbind(spR1, c(IQR(setextra1)/1.349))
  p=p+1;
}
```

C.2 Simulations Portfolio 2 assets

C.2.1 Location scale family

```
###################################
#       NORMAL DISTRIBUTION        #
###################################

sample_p = rep(NA, 1000) ;sample_k = rep(NA, 1000)
sample_r1 = rep(NA, 1000);sample_r2 = rep(NA, 1000)#;sample_r3 = rep(NA, 1000)
samplec_sdX = rep(NA, 1000) ;samplec_sdMX = rep(NA, 1000)
samplec_sdLX = rep(NA, 1000) ;samplec_sdY = rep(NA, 1000)
samplec_sdMY = rep(NA, 1000) ;samplec_sdLY = rep(NA, 1000)
sigma <- matrix(c(4,2,2,3), ncol=2)
for(i in 1:1000){
  set = rmvnorm(n=100, sigma=sigma)
  sample_p[i] = cor(set, method="pearson")[1,2] #OM cov eruit te halen
  sample_k[i] = cor(set, method="kendall")[1,2]
  sample_r1[i] = covRob(set, corr=TRUE, estim="mcd")$cov[1,2]
  sample_r2[i] = covRob(set, corr=TRUE, estim="donostah")$cov[1,2]
  samplec_sdX[i] = sd(set[,1])
  samplec_sdY[i] = sd(set[,2])
  samplec_sdMX[i] = hubers(set[,1])$s
  samplec_sdMY[i] = hubers(set[,2])$s
  samplec_sdLX[i] = mad(set[,1]/1.349)
  samplec_sdLY[i] = mad(set[,2]/1.349)
}
#COMBINATIONS FOR VAR
Zw_s1 = 0.25*samplec_sdX^2+0.25*samplec_sdY^2+0.5*sample_p #SD + PEARSON
Zw_s2 = 0.25*samplec_sdX^2+0.25*samplec_sdY^2+0.5*sample_k #SD + KENDALL
Zw_s3 = 0.25*samplec_sdX^2+0.25*samplec_sdY^2+0.5*sample_r1 #SD + covRob mcd

Zw_h1 = 0.25*samplec_sdMX^2+0.25*samplec_sdMY^2+0.5*sample_p #HUBER + PEARSON
Zw_h2 = 0.25*samplec_sdMX^2+0.25*samplec_sdMY^2+0.5*sample_k #HUBER + KENDALL
Zw_h3 = 0.25*samplec_sdMX^2+0.25*samplec_sdMY^2+0.5*sample_r1 #HUBER + covRob1

Zw_m1 = 0.25*samplec_sdLX^2+0.25*samplec_sdLY^2+0.5*sample_p #MAD + PEARSON
Zw_m2 = 0.25*samplec_sdLX^2+0.25*samplec_sdLY^2+0.5*sample_k #MAD + KENDALL
Zw_m3 = 0.25*samplec_sdLX^2+0.25*samplec_sdLY^2+0.5*sample_r1 #MAD + covRob1



#######################
#   Poluted sample   #
#######################
sample_2n = rmvnorm(n=100, sigma=sigma)
summary(sample_2n)
sd(sample_2n)
sample_2n[60,1]<-c(8)
sample_2n[60,2]<-c(-8)

par(mfrow=c(1,1))
plot.ts(sample_2n, main="")
plot(sample_2n[,1], sample_2n[,2], xlab="Series␣1", ylab="Series␣2")
points(8,-8,col = "red",pch=19)

sample2_p = rep(NA, 1000) ;sample2_k = rep(NA, 1000) ;sample2_r = rep(NA, 1000)
samplec2_sdX = rep(NA, 1000) ;samplec2_sdMX = rep(NA, 1000)
samplec2_sdLX = rep(NA, 1000) ;samplec2_sdY = rep(NA, 1000)
samplec2_sdMY = rep(NA, 1000) ;samplec2_sdLY = rep(NA, 1000)
for(i in 1:1000){
  set2n = rmvnorm(n=100, sigma=sigma)
  set2n[60,1]<-c(8*sd(set2n))
  set2n[60,2]<-c(8*sd(set2n))
  sample2_p[i] = cor(set2n, method="pearson")[1,2] #OM cov eruit te halen
  sample2_k[i] = cor(set2n, method="kendall")[1,2]
  sample2_r[i] = covRob(set2n, corr=TRUE, estim="mcd")$cov[1,2]
  sample2_r2[i] = covRob(set2n, corr=TRUE, estim="donostah")$cov[1,2]
  samplec2_sdX[i] = sd(set2n[,1])
  samplec2_sdY[i] = sd(set2n[,2])
  samplec2_sdMX[i] = hubers(set2n[,1])$s
  samplec2_sdMY[i] = hubers(set2n[,2])$s
  samplec2_sdLX[i] = mad(set2n[,1])
  samplec2_sdLY[i] = mad(set2n[,2])
}
```

```
#######################################
#        LOGISTIC DISTRIBUTION        #
#######################################

samplel_p = rep(NA, 1000) ;samplel_k = rep(NA, 1000) ;samplel_r1 = rep(NA, 1000)
samplecl_sdX = rep(NA, 1000) ;samplecl_sdMX = rep(NA, 1000)
samplecl_sdLX = rep(NA, 1000) ;samplecl_sdY = rep(NA, 1000)
samplecl_sdMY = rep(NA, 1000) ;samplecl_sdLY = rep(NA, 1000)
sigma <- matrix(c(4,2,2,3), ncol=2)
for(i in 1:1000){
  set = rmvevd(100, dep = .7, model = "log", d = 2)
  samplel_p[i] = cor(set, method="pearson")[1,2] #OM cov eruit te halen
  samplel_k[i] = cor(set, method="kendall")[1,2]
  samplel_r1[i] = covRob(set, corr=TRUE, estim="mcd")$cov[1,2]
  samplel_r2[i] = covRob(set, corr=TRUE, estim="donostah")$cov[1,2]
  samplecl_sdX[i] = sd(set[,1])
  samplecl_sdY[i] = sd(set[,2])
  samplecl_sdMX[i] = hubers(set[,1])$s
  samplecl_sdMY[i] = hubers(set[,2])$s
  samplecl_sdLX[i] = mad(set[,1])
  samplecl_sdLY[i] = mad(set[,2])
}
#COMBINATIONS FOR VAR
Zw_s1 = 0.25*samplecl_sdX^2+0.25*samplecl_sdY^2+0.5*samplel_p #SD + PEARSON
Zw_s2 = 0.25*samplecl_sdX^2+0.25*samplecl_sdY^2+0.5*samplel_k #SD + KENDALL
Zw_s3 = 0.25*samplecl_sdX^2+0.25*samplecl_sdY^2+0.5*samplel_r1 #SD + covRob

Zw_h1 = 0.25*samplecl_sdMX^2+0.25*samplecl_sdMY^2+0.5*samplel_p #HUBER + PEARSON
Zw_h2 = 0.25*samplecl_sdMX^2+0.25*samplecl_sdMY^2+0.5*samplel_k #HUBER + KENDALL
Zw_h3 = 0.25*samplecl_sdMX^2+0.25*samplecl_sdMY^2+0.5*samplel_r1 #HUBER + covRob

Zw_m1 = 0.25*samplecl_sdLX^2+0.25*samplecl_sdLY^2+0.5*samplel_p #MAD + PEARSON
Zw_m2 = 0.25*samplecl_sdLX^2+0.25*samplecl_sdLY^2+0.5*samplel_k #MAD + KENDALL
Zw_m3 = 0.25*samplecl_sdLX^2+0.25*samplecl_sdLY^2+0.5*samplel_r1 #MAD + covRob




#########################
#   Poluted sample     #
#########################
sample_2n = rmvnorm(n=100, sigma=sigma)
summary(sample_2n)
sd(sample_2n)
sample_2n[60,1]<-c(8)
sample_2n[60,2]<-c(-8)

par(mfrow=c(1,1))
plot.ts(sample_2n, main="")
plot(sample_2n[,1], sample_2n[,2], xlab="Series_1", ylab="Series_2")
points(8,-8,col = "red",pch=19)
```

```
sample2_p = rep(NA, 1000) ;sample2_k = rep(NA, 1000) ;sample2_r1 = rep(NA, 1000)
samplec2_sdX = rep(NA, 1000) ;samplec2_sdMX = rep(NA, 1000)
samplec2_sdLX = rep(NA, 1000) ;samplec2_sdY = rep(NA, 1000)
samplec2_sdMY = rep(NA, 1000) ;samplec2_sdLY = rep(NA, 1000)
for(i in 1:1000){
  set2n = rmvevd(100, dep = .7, model = "log", d = 2)
  set2n[60,1]<-c(8*sd(set2n))
  set2n[60,2]<-c(8*sd(set2n))
  sample2_p[i] = cor(set2n, method="pearson")[1,2] #OM cov eruit te halen
  sample2_k[i] = cor(set2n, method="kendall")[1,2]
  sample2_r1[i] =  covRob(set2n, corr=TRUE, estim="mcd")$cov[1,2]
  sample2_r2[i] = covRob(set2n, corr=TRUE, estim="donostah")$cov[1,2]
  samplec2_sdX[i] = sd(set2n[,1])
  samplec2_sdY[i] = sd(set2n[,2])
  samplec2_sdMX[i] = hubers(set2n[,1])$s
  samplec2_sdMY[i] = hubers(set2n[,2])$s
  samplec2_sdLX[i] = mad(set2n[,1])
  samplec2_sdLY[i] = mad(set2n[,2])
}
```

C.2.2 Factor model

```
######################
#    FACTOR MODEL    #
######################

factor_1 = rep(NA, 1000) ;factor_2 = rep(NA, 1000)
cov_f1 = rep(NA, 1000) ;cov_f2 = rep(NA, 1000)
cov_p = rep(NA, 1000) ;cov_p1 = rep(NA, 1000)
error1=rep(NA,1000);error2=rep(NA,1000)
sigma <- matrix(c(1,.2,.2,1), ncol=2)
for(i in 1:1000){
  samplef = rmvnorm(n=100, sigma=sigma)
  f1 <-rnorm(100, mean=0, sd=1)
  f2 <-rnorm(100, mean=0, sd=1)
  fit1 = lm(sample_f[,1]~f1+f2)
  fit2 = lm(sample_f[,2]~f1+f2)
  cov_p[i] = cov(samplef, method=c("pearson"))[1,2]
  samplef[60,1]<-c(8*sd(samplef))
  samplef[60,2]<-c(8*sd(samplef))
  fit3 = lm(sample_f[,1]~f1+f2)
  fit4 = lm(sample_f[,2]~f1+f2)
  cov_p1[i] = cov(samplef, method=c("pearson"))[1,2]
  cov_f1[i] = fit1$coefficients[2]*fit2$coefficients[2]+fit1$coefficients[3]*fit
  cov_f2[i] = fit3$coefficients[2]*fit3$coefficients[2]+fit4$coefficients[3]*fit
  error1[i] = summary(fit1)$coef[[4]]
  error2[i] = summary(fit2)$coef[[4]]
}
error <- matrix(c(error1, error2), nrow = 1000, ncol = 2)
summary(error)
par(mfrow=c(1,1))
```

```
plot(density(cov_f1), xlim=c(0,0.6), main="")
lines(density(cov_f2), col="blue")
lines(density(cov_p), col="green")
legend("topright", legend=c("Covariance␣unpolluted␣sample", "Covariance␣unpollut
plot(density(cov_k))
```

C.2.3 Price data

```
##############################
#        Energy/Health       #
##############################

#2 nov '98 - 12 july '17
#4706 data points
cordata = read.csv("~/Desktop/indices.csv",1)
Energy = read.csv("~/Desktop/energy.csv",1)
Health = read.csv("~/Desktop/health.csv",1)

#Convert to price changes
energy=numeric(0);
for(i in 1:4700){
  energy = rbind(energy, c(Energy[(i+1),1]-Energy[i,1]))
}
health=numeric(0);
for(i in 1:4700){
  health = rbind(health, c(Health[(i+1),1]-Health[i,1]))
}
setextra <- matrix(c(energy, health), nrow = 4700, ncol = 2)
plot(energy,health, xlab="S&P␣500␣Energy",ylab="S&P␣500␣Health␣care")
plot.ts(health, ylab="S&P␣500␣Health␣care␣price␣changes")
plot.ts(energy, ylab="S&P␣500␣Energy␣price␣changes")

#PLOT ESTIMATORS, 100 DAY LOOKBACK
pear2 = rep(NA, 4700); ken2 = rep(NA, 4700); covr2 = rep(NA, 4700)
sdX2 = rep(NA, 4700); sdY2 = rep(NA, 4700)
hubX2 = rep(NA, 4700); hubY2 = rep(NA, 4700)
madX2 = rep(NA, 4700); madY2 = rep(NA, 4700)
for(p in 1:4700){
  pear2[(p-100)] = cor(setextra[(p-100):p,], method="pearson")[1,2] #OM cov erui
  ken2[(p-100)] = cor(setextra[(p-100):p,], method="kendall")[1,2]
  covr2[(p-100)] = cov.rob(setextra[(p-100):p,],cor=TRUE)$cor[1,2]
  sdX2[(p-100)] = sd(setextra[(p-100):p,1])
  sdY2[(p-100)] = sd(setextra[(p-100):p,2])
  hubX2[(p-100)] = hubers(setextra[(p-100):p,1])$s
  hubY2[(p-100)] = hubers(setextra[(p-100):p,2])$s
  madX2[(p-100)] = mad(setextra[(p-100):p,1])
  madY2[(p-100)] = mad(setextra[(p-100):p,2])
}
#THE VAR ESTIMATIONS FOR ALL COMBINATIONS
Zw11 = 0.25*sdX2^2+0.25*sdY2^2+0.5*pear2 #SD + PEARSON 100 DAY
Zw21 = 0.25*sdX2^2+0.25*sdY2^2+0.5*ken2 #SD + KENDALL 100 DAYS
Zw31 = 0.25*sdX2^2+0.25*sdY2^2+0.5*covr2 #SD + covRob 100 DAYS
```

```
Zw41 = 0.25*hubX2^2+0.25*hubY2^2+0.5*pear2 #HUBER + PEARSON 100 DAYS
Zw51 = 0.25*hubX2^2+0.25*hubY2^2+0.5*ken2 #HUBER + KENDALL 100 DAYS
Zw61 = 0.25*hubX2^2+0.25*hubY2^2+0.5*covr2 #HUBER + covRob 100 DAYS

Zw71 = 0.25*madX2^2+0.25*madY2^2+0.5*pear2 #MAD + PEARSON 100  DAYS
Zw81 = 0.25*madX2^2+0.25*madY2^2+0.5*ken2 #MAD + KENDALL 100 DAYS
Zw91 = 0.25*madX2^2+0.25*madY2^2+0.5*covr2 #MAD + covRob 100 DAYS

####### 100 day LOOKBACK #########
par(mfrow=c(3,1))
plot.ts(Zw11,ylim=c(0,250),ylab=expression(paste(Sigma,"Zw")), main="Sample␣stan
#SD+PEARSON, 100 days
lines(Zw21, type="l",col="blue") #SD+KENDALL, 100 days
lines(Zw31, type="l",col="red") #SD+covr, 100 days
legend("topright", legend=c("Pearson", "Kendall","covRob"),col=c("black", "blue"


plot.ts(Zw41,ylim=c(0,250),ylab=expression(paste(Sigma,"Zw")), main="Huber␣stand
lines(Zw51, type="l", col="blue") #HUBER+KENDALL, 100 days
lines(Zw61, type="l",col="red")#HUBER+covRob, 100 days
legend("topright", legend=c("Pearson", "Kendall","covRob"),col=c("black", "blue"


plot.ts(Zw71,ylim=c(0,250),ylab=expression(paste(Sigma,"Zw")), main="MAD␣standar
lines(Zw81, type="l", col="blue") #MAD+KENDALL, 100 days
lines(Zw91, type="l",col="red")#MAD+covRob, 100 days
legend("topright", legend=c("Pearson", "Kendall","covRob"),col=c("black", "blue"
```