ERASMUS UNIVERSITY ROTTERDAM

Erasmus
School of
Economics

ERASMUS UNIVERSITY ROTTERDAM

Erasmus School of Economics

Chasing Black Swans: A Comparative Study of Two Random Forest Tail Risk Estimators for Financial Market Applications

January 30, 2024

Author:	Supervisor:	Co-reader:
D. van der Weerdt (456461)	prof. dr. C. Zhou	dr. P. Wan

Abstract

This research compares the performance of two new random forest tail risk estimators for financial market risk applications. These are the Extremal Random Forests method (ERF) proposed by Gnecco et al. (2022) and a method proposed by Ahmed (2022). Both methods leverage the principles of Extreme Value Theory; however, they use different techniques to extrapolate to the tail ends of a distribution. The comparison is based on three distinct simulation studies and a practical application to the Standard & Poor's 500 Index. Our findings indicate that the ERF method form Gnecco et al. (2022) is best suited for financial market applications. Nevertheless, the method from Ahmed (2022) showed promising results when combined with an eGarch-filter.

Keywords: Random Forests; Extreme Value Theory; Extremal Random Forests; Generalized Random Forests; Financial Market Risk; Value-at-Risk.

Contents

1 Introduction

1.1 Background: Financial Risk Management and combining Random Forests with Extreme Value Theory

In the domain of financial risk management, particular emphasis is placed on the occurrence of extreme losses situated in the tails of return distributions. These events can pose a significant threat to the long-term stability of financial institutions, such as banks and insurance companies. Therefore the accuracy of risk predictions is of high importance for both financial institutions and regulators.[1] It enables them to align their risk appetite with the actual risks associated with their operations. To this end, regulators and financial institutions need methods to accurately estimate the likelihood of extreme events whilst identifying the underlying risk factors contributing to these events. Two new promising methods were developed in recent literature: the method by Ahmed (2022) and the Extremal Random Forest (ERF) method by Gnecco et al. (2022). Both methods leverage the idea of combining random forests (RF) with extreme value theory (EVT).[2] Firstly, the inclusion of random forests means that the methods will be relatively flexible and able to cope with large predictor spaces allowing us to identify the risk drivers. Secondly, the inclusion of EVT allows the methods to extrapolate to extreme events beyond the available data range. Furthermore, the combination of the properties of EVT and random forests enables the incorporation of covariates through the estimation of the distribution parameters. Although both methods leverage the idea of combining random forests with EVT, there are some significant differences between the methods. The main difference lies in the way they extrapolate toward extreme events (i.e. extreme quantiles, denoted by high quantile levels $\tau \approx 1$) beyond the available data range. That is, Ahmed (2022) belongs to the class of methods that extrapolate from a given intermediate *value* (denoted by u) to a higher quantile estimate, which is similar to the methods proposed by Gilli and Këllezi (2006), Chavez-Demoulin et al. (2016), and Echaust and Just (2020). On the other hand, the ERF method developed by Gnecco et al. (2022) belongs to the class of methods that extrapolate from a given intermediate *quantile level* (denoted by τ_0) to a higher quantile estimate, so that the method can be categorized along with other quantile level extrapolation methods such as the ones from Chernozhukov (2005), Wang and Li (2013), and Velthoen et al. (2021). For instance, if we were interested in the quantile level $\tau = 0.99$, the ERF method from Gnecco et al. (2022) would

[1] We have to note here that it has historically been debated whether the use of advanced risk methods has actually made our financial markets and institutions riskier instead of more controlled. See for example Taleb (2007) or The Economist (1999). However, this discussion is beyond the scope of this paper.

[2] EVT is a branch of statistics that focuses specifically on analyzing the distribution of extreme events, rather than the overall distribution of the data. It finds applications in various disciplines, including hydrology (de Haan & Ferreira, 2006), insurance, and finance (Embrechts et al., 2013), where estimating the probability of extreme events is crucial.

extrapolate to this quantile from an intermediate lower quantile level (e.g. $\tau_0 = 0.8$). In contrast, the method from Ahmed (2022) would extrapolate from an intermediate threshold given by a certain value (e.g. $u = 20$). Consequently, this paper aims to address the question *which of the two new random forest tail risk estimation methods is best suited for financial market risk prediction?*

To assess this question, we can build upon the extensive comparative financial market analysis literature of, among others, Gencay and Selçuk (2004), Campbell (2005), Berkowitz et al. (2011), Abad et al. (2014), and Berger and Moys (2021). However, most of these papers primarily focus on backtesting the accuracy of predictions. An interesting alternative are the nine properties of good explanations developed by Robnik-Šikonja and Bohanec (2018). Nevertheless, these nine properties were not specifically developed for financial risk methods, nor do they offer a clear indication of how to measure adherence to these properties. Therefore, we combine the extensive literature on backtesting with a subset of the properties of good explanations as described by Robnik-Šikonja and Bohanec (2018).

In our research, the assessment of these properties will be conducted along three distinct simulation studies and an application using real financial market data. To ensure a fair and robust comparison, the first two simulations will be emulations of the ones developed by Ahmed (2022) and Gnecco et al. (2022). These simulations should establish a benchmark of the methods' performance on their "home turf." Subsequently, we will conduct a simulation study specifically designed to emulate financial market data. Furthermore, we will compare the performance of the methods for the widely recognized S&P500 index. This multi-faceted comparison should provide insights into the methods' effectiveness and applicability in financial risk management.

1.2 Tail Risk Modelling

Tail risk modelling involves estimating extreme quantiles (i.e. $\tau \approx 1$) of a response variable $Y \in \mathbb{R}$ based on a set of covariates $X \in \mathbb{R}^p$. In this paper, our objective is to estimate the conditional Value-at-Risk (VaR) of financial losses Y for high quantile levels τ, denoted as $VaR_x(\tau)$ with $\tau \approx 1$. Essentially, this is the same as a quantile estimation, as the VaR is by definition a quantile. However, estimating high quantile levels often poses challenges due to the limited availability of observations in the tail ends of the data range. (Taleb, 2007) illustrated this problem with the following example: *"Before the discovery of Australia, people in the Old World were convinced that all swans were white, an unassailable belief as it seemed completely confirmed by empirical evidence."* Similarly, traditional empirical estimators, such as the Historical Simulation method and the RiskMetrics approach, face limitations in properly incorporating the probability of losses exceeding the previously observed data range (Abad et al., 2014). Consequently, using these types of methods would lead to large biases (Gnecco et al. (2022)). Moreover, the lack of observations prevents us from making use of a large number of covariates. The predictor space required to train the model exceeds the coverage provided by the available data, and the relationship between the covariates and the parameters that describe the tail regions of the distribution are often highly nonlinear. As a result, standard quantile regression models, like the one developed by Koenker and Bassett (1978), struggle to capture the behavior of the tail under various predictor configurations, again leading to significant bias.

To overcome the challenge of extending loss estimation beyond the bounds of available data, risk managers have the option to employ asymptotically motivated approximation methods derived from extreme value theory (EVT). Some examples of EVT extrapolation methods are the aforementioned quantile level extrapolation methods from Chernozhukov (2005), Wang and Li (2013), Velthoen et al. (2021), and now Gnecco et al. (2022), or the intermediate value extrapolation methods from Gilli and Këllezi (2006), Chavez-Demoulin et al. (2016), Echaust and Just (2020), and now Ahmed (2022).

Furthermore, to make full use of the available covariates, we need methods capable of handling complex and large predictor spaces with limited training data. Over the years, several approaches have been developed to address high-dimensional predictor spaces. For example, Farkas et al. (2021) use regression trees to adapt to larger predictor dimensions. Similar to Gnecco et al. (2022) and Ahmed (2022), Meinshausen and Ridgeway (2006), Athey, Tibshirani, and Wager (2019), and Staudt and Wagner (2021) use random forests to be able to handle this issue. Additionally, the GBEX method proposed by Velthoen et al. (2021) uses gradient boosting, which also showed promising results.

Among these approaches, the forest-based methods stand out due to their ability to perform well with a relatively small amount of observations for tuning, while also having well-understood statistical properties (Athey et al., 2019). Moreover, using a forest-based method also addresses the nonlinear relation between the covariates and the parameters describing the tail ends of the distribution.

As both methods of interest combine the benefits of EVT and random forests, this paper contributes to the rich literature of tail risk forecasting by conducting a comprehensive and all-round comparison between two promising tail risk estimation methods: the ERF method proposed by Gnecco et al. (2022) and the method introduced by Ahmed (2022). To the best of our knowledge, no prior comparison has been made between these two methods, thus further advancing our understanding of their capabilities and limitations.

1.3 Research Outline

This research sets up a comprehensive comparison between the ERF method (Gnecco et al., 2022) and the method from Ahmed (2022) through multiple simulation studies and an application on real financial market data. Primarily we found that that the ERF methodology is best suited for financial market risk prediction. However, the method from Ahmed (2022) exhibited promising results when combined with an eGarch(1,1) filtering technique.

The remainder of this paper is structured as follows: Section 2 provides a description of the methods employed in this research. In particular, this section focuses on the explanation of the methods proposed by Ahmed (2022) and Gnecco et al. (2022). Thereafter, Section 3 presents an examination of the three simulation studies conducted in this study. Additionally, in this section, we compare the performance of the methods using the simulated data. Moving forward, Section 4 clarifies the utilized stock market data and offers insights into the used data pre-processing techniques. Lastly, in Section 5 we present our conclusion based on the findings obtained in the previous sections. Furthermore, this section provides suggestions for further research, thereby contributing to the ongoing discourse in the field.

2 Methodology

As discussed in the Introduction, two models have been introduced that combine the advantages of extreme value theory and random forests for estimating extreme quantiles. This section aims to clarify the main variable of interest we are modelling, give some background on Extreme Value Theory, explain the underlying procedure of the methods, and describe the methods used for the comparative analysis.

2.1 VaRiable of Interest

First, we start by defining the dependent variable we are trying to model. As alluded to in the Introduction, we are interested in estimating the greatest possible daily outcome of the financial (log) loss returns $Y \in \mathbb{R}$ at confidence levels close to one (i.e. quantile level $\tau \approx 1$), conditional on certain informative covariates $X \in \mathbb{R}^p$. This is by definition the Value-at-Risk (VaR) conditional on X at high confidence levels, denoted by $VaR_X(\tau)$. VaR is a commonly used risk metric in financial applications and can be understood more simply as a quantile of the loss distribution (Berger & Moys, 2021). This means that the idea behind our variable of interest is given by ”the highest potential loss that is not surpassed with a specified probability". Formally, we use:

$$VaR_X(\tau) = \inf\{y \in \mathbb{R} : \mathbb{P}(Y > y|X = x) \leq 1 - \tau\} = \inf\{y \in \mathbb{R} : F_Y(y, x) \geq \tau\}. \tag{2.1}$$

In other words, the VaR of a financial market portfolio at confidence level τ ($\tau \in (0, 1)$), is the minimum value y for which the probability of the loss Y surpassing y conditional on $X = x$ is no larger than $1 - \tau$.

2.2 Extreme Value Theory

Within EVT there are two primary approaches for selecting empirical data points to analyze the distribution of extreme events (Ahmed, 2022): the Block Maxima (BM) approach and the Peak Over Threshold (POT) approach. The former divides the data into uniformly sized blocks and uses the maximum value within each block, while the latter uses all observations that exceed a certain threshold. Both (Gnecco et al., 2022) and (Ahmed, 2022) use the more modern POT approach as it is considered more efficient (McNeil et al., 2015). Consequently, we will only consider the POT approach in this paper.

The fundamental concept of the POT method is to characterize the distribution of exceedances beyond a certain threshold. That is, if we again use the financial (log) loss returns Y and assume it has distribution function F. The POT method would use a threshold u to define the excess distribution function of the threshold exceedances $Z = Y - u$:

$$F_u(z) = \mathbb{P}(Y - u \leq z|Y > u) = \frac{F(z + u) - F(u)}{1 - F(u)}, \quad 0 \leq z \leq y_F - u, \tag{2.2}$$

with $y_F \leq \infty$ as the right endpoint of the distribution function F.

The POT method then uses the Pickands-Balkema-De Haan theorem (Balkema & De Haan, 1974) which states that, under the mild assumption that the loss distribution is in the maximum domain of attraction of some extreme value distribution (which most continuous distributions in statistics are (McNeil et al., 2015)), $F_u(z)$ can be modelled by a Generalized Pareto Distribution (GPD) $G(z; (\sigma, \xi))$. The distribution function of the GPD for the heavy-tailed case is given by:

$$G(z; \theta) = 1 - \left(1 + \frac{\xi}{\sigma} z\right)_+^{-1/\xi}, \quad z > 0, \tag{2.3}$$

where $\theta = (\sigma, \xi) \in (0, \infty) \times \mathbb{R}$ contains the scale and shape parameter, respectively. When $\xi = 0$ the distribution is light-tailed and when $\xi < 0$ the distribution is constrained by a finite upper endpoint (Gnecco et al., 2022).

The GPD distribution is used in conjunction with Bayes' theorem, where Bayes' theorem is employed to obtain the probability $\mathbb{P}(Y > y)$ that the loss Y exceeds a high threshold y (given $y > u$). This combination is used to obtain an approximate of a quantile $\tau \to 1$ of loss Y:

$$Q(\tau) \approx Q(\tau_0) + \frac{\sigma}{\xi}\left[\left(\frac{1-\tau}{1-\tau_0}\right)^{-\xi} - 1\right], \tag{2.4}$$

where $\mathbb{P}(Y > y) = 1 - \tau$, $\mathbb{P}(Y > u) = 1 - \tau_0$, and $Q(\tau_0) := F_Y^{-1}(\tau_0)$ denotes the intermediate quantile at level $\tau_0 < \tau$.

In practice, this means that we have to estimate the scale σ and shape ξ parameters from the empirical data to extrapolate to extreme events.

2.3 Extremal Random Forest - (Gnecco et al., 2022)

The Extremal Random Forest (ERF) method was introduced by Gnecco et al. (2022). As mentioned in the Introduction, this method combines the extrapolation techniques from extreme value theory and the flexibility of random forests for extreme quantile estimation. This section gives a detailed description of the method and explains the way in which it uses the weights from a quantile random forest to calculate the parameters of interest in EVT extrapolation conditional on a set of predictors.

2.3.1 Extreme Quantile Extrapolation

The method uses EVT to extrapolate the VaR conditional on covariates $X = x$ with probability levels close to one, by using an altered version of equation (2.4) from Balkema and De Haan (1974) and Pickands III (1975). This altered equation is again based on the GPD, which, under the mild regularity assumption on the tail of $Y|X = x$ that the distribution function should be continuous and in the domain of attraction of an extreme value distribution, can be used to make approximations of values beyond a certain threshold (Gnecco et al., 2022):

$$VaR_x(\tau) \approx VaR_x(\tau_0) + \frac{\sigma(x)}{\xi(x)}\left[\left(\frac{1-\tau}{1-\tau_0}\right)^{-\xi(x)} - 1\right], \tag{2.5}$$

where $VaR_x(\tau_0)$ is an intermediate quantile with $\tau_0 < \tau$, that can be accurately estimated by classical quantile regression methods, $\sigma(x)$ denotes the conditional scale (with $\sigma(x) \in (0, \infty) \times \mathbb{R}$) of the GPD, and $\xi(x)$ the conditional shape (with $\xi(x) \in (0, \infty) \times \mathbb{R}$) parameter of the GPD.

To make full use of the advantages of EVT, the intermediate quantile τ_0 should be chosen carefully. It should be high enough to ensure the accuracy of the approximation in equation (2.5) and, at the same time, sufficiently low to allow accurate estimation of $VaR_x(\tau_0)$ through classical quantile regression methods.

After choosing τ_0, the ERF algorithm estimates $\widehat{VaR_x}(\tau_0)$ with the Generalized Random Forest (GRF) with a quantile loss from Athey et al. (2019) (see Section 2.3.3) as its quantile regression method (although one could theoretically choose any method). This outcome is then used to calculate the exceedances $Z_i = (Y_i - \widehat{VaR_x}(\tau_0))_+$, $i = 1, ..., n$ with n denoting the number of observations. Z_i is used in the next part when we estimate the GPD parameter vector $\theta(x) = (\sigma(x), \xi(x))$.

2.3.2 Scale and Shape Parameter Estimation

After estimating $\widehat{VaR_x}(\tau_0)$, the only missing elements from equation (2.5) are the scale and shape parameters $\sigma(x)$ and $\xi(x)$. When using EVT, $\theta(x)$ can be estimated by Maximum Likelihood (ML), where the negative log-likelihood contribution of the ith exceedance (i.e. Z_i) is calculated via the following equation:

$$\ell_\theta(Z_i) = \log\,\sigma + \left(1 + \frac{1}{\xi}\right) \log\left(1 + \frac{\xi}{\sigma} Z_i\right), \quad \theta \in (0, \infty) \times \mathbb{R}, \tag{2.6}$$

if $Z_i > 0$, and zero otherwise.

Similarly, the ERF algorithm also uses this equation (see Algorithm 1). However, as we are interested in the parameters conditional on $X = x$, the method incorporates the similarity weights $w_n(x, X_i)$ from a GRF (Athey et al., 2019) (see Section 2.3.3) in order to find $\hat{\theta}(x)$. It does so via the following weighted negative log-likelihood equation:

$$L_n(\theta; x) = \sum_{i=1}^{n} w_n(x, X_i)\ell_\theta(Z_i)\mathbb{1}\{Z_i > 0\}, \quad x \in \mathcal{X} \subset \mathbb{R}^p, \tag{2.7}$$

with $\mathcal{X}$ compact, and $\ell_\theta(Z_i)$ as in equation (2.6).

Gnecco et al. (2022) point out that the parameter space $\theta(\mathcal{X}) = \{\vartheta \in (0, \infty) \times \mathbb{R} : \vartheta = \theta(x)$ for some $x \in \mathcal{X}\}$ is unknown in practice, and that there is no guarantee for a global optimum. Therefore Gnecco et al. (2022) follow Bücher and Segers (2017) by defining:

$$\hat{\theta}(x) = \arg\min_{\theta \in \Theta} L_n(\theta; x), \tag{2.8}$$

with an arbitrarily large compact set $\Theta \subset (0, \infty) \times (0, \infty)$ such that $\theta(\mathcal{X}) \subset \text{Int}\ \Theta$ and $L_n(\theta; x)$ as in equation (2.7).

If multiple minima are found, the algorithm follows the lexicographic order. This means it will first sort the minimizers from small to large via the scale parameter, and sort the minimizers with equal scale parameters from small to large via the shape parameter, after which it uses the first minimizer as $\hat{\theta}(x)$.

As Gnecco et al. (2022) correctly point out, the estimation of the shape parameter ξ is both crucial for the quality of the output and remarkably challenging. To this end, multiple penalization methods were built in order to reduce the variance of the ξ estimator at the cost of increased bias. In particular, Gnecco et al. (2022) want to penalize the variation of $\xi(x)$ over the entire predictor space $\mathcal{X}$. That is, they reduce $\hat{\xi}(x)$ towards a constant shape parameter ξ_0. We will follow Gnecco et al. (2022) in their configuration of $\xi_0 = \hat{\xi}$ attained by minimizing equation (2.7) with similarity weights set to $w_n(x, y) = 1$ for all $x, y \in \mathcal{X}$. In doing so they change equation (2.8) by adding a penalization parameter λ as follows:

$$\hat{\theta}(x) = \arg\min_{\theta \in \Theta} \frac{1}{(1-\tau_0)} L_n(\theta; x) + \lambda(\xi - \xi_0)^2, \tag{2.9}$$

with penalization parameter $\lambda \geq 0$. This penalization scheme can be interpreted as follows: when λ is large, the model will become simpler as it disallows large variations in the shape parameter, and when λ is small the model becomes more complex with little penalization on a varying shape parameter ξ across the predictor space $\mathcal{X}$.

2.3.3 Similarity Weights

To estimate high quantile levels of a distribution conditional on a set of covariates $X = x$, we would normally need training observations around $X = x$. By contrast, for tail risk modelling we need to extract information from the training observations where $X_i \neq x$. Logically, one can learn more from the training observations that have covariates close to x than the ones that have covariates that are further distanced from x in the predictor space. Hence, we would like to assign 'similarity weights' to our training observations relative to a new instance $X = x$. To this end, we need a similarity weight function $w_n(x, X_i)$ that estimates the relevance of each training observation X_i to the estimation of $VaR_x(\tau)$. The classical quantile regression estimator can then use this weight function through:

$$\widehat{VaR}_x(\tau) = \arg\min_{q \in \mathbb{R}} \sum_{i=1}^{n} w_n(x, X_i)\rho_\tau(Y_i - q), \tag{2.10}$$

with $\rho_\tau(c) = c(\tau - \mathbb{1}\{c < 0\})$, $c \in \mathbb{R}$ as the quantile loss function (Koenker & Bassett, 1978).

The GRF method (Athey et al., 2019) builds on the work from Meinshausen and Ridgeway (2006), who use estimator (2.10) with a weight function $w_n(x, X_i)$ obtained from a random forest regression. That is, they build a forest containing B trees with the training observations, and for each tree $b \in \{1, ..., B\}$ they drop all the training observations X_i and the new instance x down the tree and define $L_b(x)$ as the set of training observations that end in the same 'leaf' as x. The weight given to the ith training observation X_i is then:

$$w_n(x, X_i) = \frac{1}{B} \sum_{b=1}^{B} \frac{\mathbb{1}\{X_i \in L_b(x)\}}{|L_b(x)|}, \tag{2.11}$$

where the sum of the weights of all training observations is equal to one.

The same process is used in the GRF method by Athey et al. (2019), however they use a different splitting scheme when constructing the forest. That is, Meinshausen and Ridgeway (2006) use the standard CART regression splits introduced by Gordon et al. (1984) which means the trees are grown by minimizing the mean squared error loss, causing the weight function to favour those observations where $E[Y|X = X_i] \approx E[Y|X = x]$ instead of focusing on the entire conditional distribution. Instead, Athey et al. (2019) use a splitting scheme based on minimizing the quantile loss function. This leads to similarity weights that better capture the heterogeneity in the relation between Y and X. Gnecco et al. (2022) fit a GRF on the training data to obtain the similarity weight function $w_n(.,.)$, which is used in equation (2.7).

Furthermore, Gnecco et al. (2022) also use the GRF method to estimate $\widehat{VaR}_x(\tau_0)$, where the weight function is used as in equation (2.10). As stated in Section 2.3.1, the latter can also be done using a different quantile regression method, however, as the GRF requires little tuning and is a proper method for the high-dimensional setup, Gnecco et al. (2022) use this method for the intermediate VaR estimate.

After fitting the GRF method on the training data, we obtain the similarity weight function used in equation (2.8) and can estimate the intermediate quantile $\widehat{VaR}_x(\tau_0)$. This means we have all the necessary ingredients to estimate $\widehat{VaR}_x(\tau)$ as in equation (2.5).

2.3.4 ERF Algorithm

All the previous steps are used to get to the ERF algorithm. Described in Algorithm 1, the procedure consists of two main steps, called 'ERF-Fit' and 'ERF-Predict'. Firstly, ERF-Fit is used to estimate both the similarity weights as described in Section 2.3.3 and the intermediate $\widehat{VaR}_x(\tau_0)$ as described in Section 2.3.1. Afterward, ERF-Predict uses the output from ERF-Fit to calculate the exceedances Z_i as described in Section 2.3.1, which in turn is used to estimate the conditional scale and shape parameters $\hat{\theta}(x) = (\hat{\sigma}(x), \hat{\xi}(x))$ as described in Section 2.3.2. Having estimated $\widehat{VaR}_x(\tau_0)$ and $\hat{\theta}(x) = (\hat{\sigma}(x), \hat{\xi}(x))$, the algorithm then returns $\widehat{VaR}_x(\tau)$ as described by equation (2.5).

Algorithm 1 Extremal Random Forest (ERF)

```
Require: Training data $\mathcal{D} = \{(X_i, Y_i)\}_{i=1}^n$, predictor value $x \in \mathbb{R}^p$, intermediate and extreme
    quantile levels $\tau_0$ and $\tau$ (with $\tau_0 < \tau$), and the vector of hyperparameters $\alpha$.
procedure ERF-Fit($\mathcal{D}, \tau_0, \alpha$)
    $w_n(\cdot,\cdot) \leftarrow \text{GRF}(\mathcal{D}, \alpha)$
    $\widehat{VaR}_{\cdot}(\tau_0) \leftarrow \text{QuantileRegresssion}(\mathcal{D})$
    output erf $\leftarrow [\mathcal{D}, w_n(\cdot,\cdot), \widehat{VaR}_{\cdot}(\tau_0)]$
procedure ERF-Predict(erf, $x, \tau$)
    $\text{Z}_i \leftarrow (Y_i - \widehat{VaR}_{X_i}(\tau_0))_+$, with $i = 1, ..., n$
    $\hat{\theta}(x) \leftarrow \arg\min_\theta \frac{1}{(1-\tau_0)} L_n(\theta; x) + \lambda(\xi - \xi_0)^2$ as in (2.9)
    output $\widehat{VaR}_x(\tau)$ as in (2.5)
```

2.3.5 Hyperparameter Tuning

To tune the hyperparameters of the random forest, we will apply the cross-validation scheme used by Gnecco et al. (2022). In their research, they make use of the negative log-likelihood contribution (or deviance) from the GPD (as shown in equation (2.6)) as the relevant metric for cross-validation. They do not use the quantile loss, as this is not a reliable metric for large τ values. Then, the scheme entails a random division of the training data into M equally sized subsets $N_1, ..., N_M$, on which the sequence of J hyperparameter configurations $\alpha_1, .., \alpha_J$ will be fitted through the ERF-Fit procedure of Algorithm 1 on training data (X_i, Y_i), $i \notin N_m$ for all M folds. Following the ERF-Fit procedure of Algorithm 1, we use these erf-objects to estimate the parameter vector $\hat{\theta}(X_i; \alpha_j)$ on the validation data (X_i, Y_i), $i \in N_m$ for all M folds. This is then used to calculate the cross-validation error for each α_j configuration by:

$$CV(\alpha_j) = \sum_{m=1}^{M} \sum_{i \in N_m} \ell_{\hat{\theta}(X_i;\alpha_j)}(Z_i) \mathbb{1}\{Z_i > 0\}, \tag{2.12}$$

with $\theta \mapsto \ell_\theta(z)$ representing the deviance of the GPD as described in Section 2.3.2, and with exceedances Z_i as defined in Section 2.3.1. The optimal hyperparameter configuration α^* is chosen as the configuration that minimizes equation (2.12).

In order to make the scheme computationally feasible, Gnecco et al. (2022) start by fitting the intermediate quantile function $x \mapsto \widehat{VaR}_x(\tau_0)$ on the entire training data set. This is followed by the estimation of the similarity weight function $(x, y) \mapsto w_n(x, y)$ through relatively small forests of 50 trees.

Gnecco et al. (2022) mainly focuses on tuning the minimum node size (denoted by $\kappa \in \mathbb{N}$), and the penalization parameter λ (with λ as in Section 2.3.2). We follow them by setting all other parameters to a default setting, whilst tuning the κ and λ parameters according to the aforementioned cross-validation scheme.

2.3.6 Changes to ERF Algorithm

As mentioned in the Introduction, we intend to compare the feasibility of the methods for financial market risk modelling. In general, it is accepted that financial market data is heavy-tailed, see for example the research from Mandelbrot (1963) or the discussion paper of Bradley and Taqqu (2003). However, the ERF algorithm as described in this section makes use of the GPD, which not only accommodates heavy-tailed data (when $\xi > 0$) but also light-tailed data (when $\xi = 0$) and data with finite upper-end points (when $\xi < 0$) (Gnecco et al., 2022). Consequently, the algorithm does not specifically target heavy-tailed and thus financial market data in the same way the method from Ahmed (2022) does. Therefore, in order to make a fair comparison, we will make some adjustments to the ERF method to accommodate for the heavy-tailed characteristics of financial market data.

To target heavy-tailed data, we assume that $\xi > 0$. Then the exceedances $Y > VaR_x(\tau_0)$ belong to the maximum domain of attraction (MDA) of the Fréchet distribution. Following McNeil et al. (2015), this means that the distribution is approximately Pareto and hence we can use the inverse of the standard form Hill tail estimator (also known as the Weissman estimator (Weissman, 1978)) given by equation (5.24) on page 160 of (McNeil et al., 2015), given by:

$$\widehat{VaR}_x(\tau) = \widehat{VaR}_x(\tau_0) \left(\frac{1-\tau_0}{1-\tau}\right)^{\hat{\xi}(x)}. \tag{2.13}$$

In addition, given that we are dealing with a special case of the GPD, we can follow Theorem 1 from Gnecco et al. (2022) and set the scale parameter to $\sigma(x) = \xi(x)VaR_x(\tau_0)$. Consequently, we can find the equation for the negative log-likelihood contribution of the ith exceedance in:

$$\ell_\xi(Z_i) = \log(\xi) + (1 + \frac{1}{\xi})\log\left(1 + \frac{Z_i}{\widehat{VaR}_x(\tau_0)}\right), \quad \xi \in (0, \infty) \times \mathbb{R}, \tag{2.14}$$

if $Z_i > 0$, and zero otherwise.

Neglecting the penalty parameter λ for a moment, it is clear that there is an explicit solution to these first-order conditions and so there is no need for optimization as in equation (2.8). To this end, Gnecco et al. (2022) define the *random forest Hill estimator* given by:

$$\hat{\xi}_H(x) = \frac{n}{k}\sum_{i=1}^{n} w_n(x, X_i)\mathbb{1}\{Z_i > 0\}\log\left(1 + \frac{Z_i}{\widehat{VaR}_x(\tau_0)}\right), \tag{2.15}$$

with $k = n(1 - \tau_0)$.

Nevertheless, it is different when we use the penalization parameter λ, as there is no singular solution and solving explicitly may lead to complex solutions. Therefore, when using the penalization parameter, we estimate $\xi(\hat{x})$ by plugging equation (2.14) into equation (2.7).

Beyond the mentioned alterations, all other aspects remain unchanged when we apply these ERF configurations tailored to the heavy-tailed Pareto distribution.

2.4 Conditional Value-at-Risk using Random Forest - (Ahmed, 2022)

The tail risk model proposed by Ahmed (2022) is based on similar principles as the Extremal Random Forest model from Gnecco et al. (2022) (i.e. combining EVT and random forests). However, there are some key differences which we will explain in this sub-chapter.

2.4.1 Differences compared to ERF Algorithm

One of the differences in the methodologies is that Ahmed (2022) specifically focuses on data with a heavy-tailed distribution function F that belongs to the maximum domain of attraction (MDA) of an extreme value distribution H_ξ. Having a heavy-tailed distribution in the MDA of H_ξ means that the shape parameter (also called the extreme value index) should be positive ($\xi > 0$).

This assumption allows the use of Theorem 1.2.1 from de Haan and Ferreira (2006) which states that the assumption on distribution function F of $\xi > 0$ and $F \in MDA(H_\xi)$ is true if and only if:

$$\lim_{t \to \infty} \mathbb{P}(Y > ty | Y > t) = y^{-1/\xi}. \tag{2.16}$$

Another distinction is that Ahmed (2022) choose to extrapolate from a fixed-value threshold u instead of using a fixed probability threshold τ_0 as done by (Gnecco et al., 2022). Again following Theorem 1.2.1 from de Haan and Ferreira (2006), using a high enough threshold u allows the use of the aforementioned Hill tail estimation formula from EVT (McNeil et al., 2015) to extrapolate to a VaR quantile beyond the available data range:

$$\widehat{VaR}(\tau) = u \left(\frac{\hat{g}}{1 - \tau} \right)^{\hat{\xi}}. \tag{2.17}$$

with $\hat{\xi}$ again as the shape parameter and $\hat{g} = \mathbb{P}(\widehat{Y > u})$ as an estimate for the exceedance probability of Y being larger than threshold u.

One could now calculate the unconditional $\widehat{VaR}(\tau)$ by using the empirical exceedance probability for $\hat{g}$ and a Hill estimator for $\hat{\xi}$. However, as Ahmed (2022) want to include a conditional dependence $Y|X = x$, they define $\xi(x)$ as the conditional shape parameter (more detailed description in Section

2.4.2) and $g(x) = \mathbb{P}(Y \geq u|X = x)$ as the conditional probability of $Y|X = x$ exceeding u. Equation (2.17) then changes to:

$$\widehat{VaR}_x(\tau) = u\left(\frac{\hat{g}(x)}{1-\tau}\right)^{\hat{\xi}(x)}, \tag{2.18}$$

with $\hat{g}(x) = \widehat{\mathbb{P}(Y > u|X = x)}$ denoting the estimated conditional probability of exceeding u.

2.4.2 Estimation of $\xi(x)$

To estimate the VaR, it is necessary to estimate $\xi(x)$. With this objective in mind, Ahmed (2022) derived the following approximation from EVT:

$$\xi(x) \approx E\left(\log\left(\frac{Y}{u}\right)\middle|Y > u, X = x\right) = E\left(S|X = x\right), \tag{2.19}$$

with $S_i = \log\left(\frac{Y_i}{u}\middle|Y_i > u\right)$ similar to Z_i as defined in Section 2.3.1.

Ahmed (2022) points out that $E\left(S|X = x\right)$ can be estimated using a random forest regression algorithm. In the algorithm, they use a splitting process that maximizes the decrease of the least squared error. That is, if we create B regression trees T_r (r indicating regression tree) in a forest, then for each tree $b \in \{1, ..., B\}$ the random forest regression algorithm constructs an architecture Θ_b in relation to splits and nodes. For an observation i, with covariates x_i, each regression tree $T_r(.; \Theta_b)$ then gives a local estimate for the shape parameter $\hat{\xi}(x_i) = T_r(x_i; \Theta_b)$. Using the standard random forest procedure pioneered by Breiman (2001), we then aggregate the outcome of all the trees to obtain our localized conditional shape parameter estimate. For the general case of $X = x$, the estimator for the conditional shape parameter then becomes:

$$\hat{\xi}(x) = \frac{1}{B}\sum_{b=1}^{B} T_r(x; \Theta_b). \tag{2.20}$$

2.4.3 Estimation of $g(x)$

The conditional probability $g(x)$ of Y exceeding the threshold u given $X = x$ is estimated by first building a random forest classification model, after which they follow Niculescu-Mizil and Caruana (2005) in calibrating the estimates using the Platt Calibration method (Platt et al., 1999) to produce robust probability estimates.

Ahmed (2022) use the random forest classification model to estimate the exceedance probability

$g(x)$ via the conditional expectation $E(V_i|X = x)$, with:

$$V_i = \begin{cases} 1 \text{ if } Y_i > u, \\ 0 \text{ if } Y_i \leq u \end{cases} .$$

In the splitting process, the best split is defined as the split that maximizes the Gini gain (Ahmed, 2022). The outcome of the random forest classification model then becomes:

$$\hat{f}(x) = \frac{1}{B}\sum_{b=1}^{B} T_c(x; \Theta_b), \tag{2.21}$$

with $T_c(x; \Theta_b)$ representing the local prediction of a single classification tree analogous to the description in Section 2.4.2 of the local prediction of a single regression tree $T_r(x; \Theta_b)$.

As $\hat{f}(x)$ should not be used as a probability, Ahmed (2022) use the Platt Calibration method as described by Niculescu-Mizil and Caruana (2005) to obtain calibrated probabilities for $\mathbb{P}(V = 0|X = x)$. That is, we estimate the regression:

$$\mathbb{P}(V = 0|X = x) = \frac{1}{1 + \exp(\beta_0 f(x) + \beta_1)},$$

with β_0 and β_1 as the parameters to be estimated by minimizing:

$$(\hat{\beta}_0, \hat{\beta}_1) = \min_{(\beta_0, \beta_1)} -\sum_{i=1}^{n} V_i \log(p_i) + (1 - V_i)\log(1 - p_i), \tag{2.22}$$

with $p_i = \frac{1}{1+\exp(\beta_0 \hat{f}(x)+\beta_1)}$.

Then to estimate $g(x)$, we use $\mathbb{P}(V = 1|X = x) = 1 - \mathbb{P}(V = 0|X = x)$. Hence, the equation used for the conditional probability of Y exceeding u given $X = x$, is defined as:

$$\hat{g}(x) = \frac{\exp(\hat{\beta}_0 \hat{f}(x) + \hat{\beta}_1)}{1 + \exp(\hat{\beta}_0 \hat{f}(x) + \hat{\beta}_1)}. \tag{2.23}$$

We follow Ahmed (2022), in using the Brier score (Brier, 1950) to evaluate the calibrated probabilities $\hat{g}(x)$. That is, we use it for hyperparameter tuning (see Section 2.4.5). See Section 2.5.1 for a more detailed description.

2.4.4 Algorithm

Again, all the previous steps are used to get to the algorithm developed by Ahmed (2022). Different from Algorithm 1, Algorithm 2 only consists of a ’Predict’ step which consists of the three steps that were described in Sections 2.4.2 and 2.4.3. These are, the random forest regression used to estimate $\hat{\xi}(x)$, the random forest classification used to calculate $\hat{f}(x)$ which are calibrated in the third step with the Platt Calibration method to obtain $\hat{g}(x)$. Using equation (2.18) and $\hat{\xi}(x)$ and $\hat{g}(x)$, the algorithm then returns $\widehat{VaR}_x(\tau)$.

Algorithm 2 Ahmed (2022)

```
Require: Training data $\mathcal{D} = \{(X_i, Y_i)\}_{i=1}^n$, predictor value $x \in \mathbb{R}^p$, intermediate threshold $u$,
    extreme quantile level $\tau$, and the vector of hyperparameters $\alpha$.
procedure PREDICT($\mathcal{D}, u, \alpha$)
    $\hat{\xi}(x) \leftarrow$ RF regression $(\mathcal{D}, u, \alpha)$
    $\hat{f}(x) \leftarrow$ RF classification $(\mathcal{D}, u, \alpha)$
    $\hat{g}(x) \leftarrow$ Platt Calibration method $(\mathcal{D}, \hat{f}(x))$
    output $\widehat{VaR}_x(\tau)$ as in (2.18)
```

Given that the modifications to the ERF method (see Section 2.3.6) were specifically designed to adapt it to the heavy-tailed Pareto environment for which the method of Ahmed (2022) was developed, there is no need to make any alterations to the methodology presented in Ahmed (2022). Hence, we can directly use Algorithm 2 as is.

2.4.5 Hyperparameter Tuning

Similar to Gnecco et al. (2022), Ahmed (2022) also limit their hyperparameter tuning to the minimum node size. Hence we will make use of the cross-validation scheme as described in Section 2.3.5, without using the cross-validation error from equation (2.12), Instead, for the estimation of $\hat{g}(x)$, we will follow Ahmed (2022) by using the Brier score (Brier, 1950) to find the 'optimal' minimum node size for the random forest classification model. Furthermore, using this 'optimal' minimum node size setting for the classification forest, we follow the same cross-validation scheme to find the 'optimal' minimum node size for the regression forest by using the Proportion of Failures test (PoF), which is discussed later in Section 2.5.1.

Lastly, we follow Ahmed (2022) in their selection of the appropriate threshold u by means of a Hill plot. That is, they create a Hill plot using all the training data and choose from a wide range of values.

2.5 Methods of Comparison

As we mentioned in the Introduction, we will combine the accuracy measures of the extensive literature on VaR backtesting (see Gencay and Selçuk (2004), Campbell (2005), Berkowitz et al. (2011), Abad et al. (2014), and Berger and Moys (2021)) with four of the nine properties of good explanations as described by Robnik-Šikonja and Bohanec (2018). These four properties can be described as follows:

1. Prediction Accuracy: the ability of the model to make accurate predictions at different quantile levels.
2. Robustness: the resilience to outliers and noise variables.
3. Consistency: the consistency of the accuracy under changing market conditions.
4. Importance: the capacity of the model to communicate the key variables that influenced the predicted outcome.

In this section, we will explain the individual components.

2.5.1 Prediction Accuracy

When comparing the accuracy of the two methods, we have to make a distinction between the real data application and the simulation studies. That is, for the simulation studies we know the actual VaR at any quantile level τ because we know the data generating process (DGP), whereas for the real data application we have to make inferences from the empirical sample at hand.

For the simulation studies we will follow Gnecco et al. (2022) in using the square root of the Integrated Squared Error (ISE) and the Mean ISE (MISE). Specificallly, the ISE is defined as:

$$\text{ISE} = \frac{1}{n}\sum_{i=1}^{n}\left(\widehat{VaR}_{x_i}(\tau) - VaR_{x_i}(\tau)\right)^2, \tag{2.24}$$

where $x \to VaR_x(\tau)$ is the true VaR inferred from the DGP. The ISE is computed based on one simulation, whereas the MISE is obtained by averaging the ISE over a number of repeated simulations.

To evaluate the accuracy for the real data application, we will make use of the 'hit rate' (Christoffersen, 1998) given by:

$$H_i(\tau) = \begin{cases} 1 \text{ if } Y_i > VaR_{x_i}(\tau) \\ 0 \text{ if } Y_i \leq VaR_{x_i}(\tau) \end{cases}. \tag{2.25}$$

This indicator function simply counts the number of times the return variable has a value lower than the VaR estimate. Using the hit rate allows us to evaluate two important properties of a good VaR measure as mentioned by Christoffersen (1998):

1. The Unconditional Coverage Property: the number of VaR violations (i.e. $\sum_{i=1}^{n} H_i(\tau)$) should be close to the observed number of expected VaR violations (i.e. $n \times (1-\tau)$).

2. The Independence property: observed violations should not have any explanatory power over future violations.

When both properties are satisfied, the VaR forecasts are considered to have correct conditional coverage. This means the hit rate is Bernoulli distributed with the probability equal to $1-\tau$ (Berger & Moys, 2021):

$$H_i(\tau) \overset{i.i.d.}{\sim} Bernoulli(1-\tau). \tag{2.26}$$

The first property can be compared using the observed violation ratio $VR(\tau)$:

$$VR(\tau) = \frac{1}{n}\sum_{i=1}^{n} H_i(\tau), \tag{2.27}$$

where n is the number of observations used for backtesting.

This ratio is used to test $H_0 : E[H(\tau)] = 1-\tau$ by means of the aforementioned PoF-test (see Section 2.4.5). This test is one of the standard VaR backtesting tests (Ahmed, 2022) and uses the likelihood-ratio (LR) statistic (Kupiec, 1995) as:

$$\text{LR}_{PoF} = -2\ln\left(\frac{(1-\tau)^{n-r}\tau^{r}}{(1-VR(\tau))^{n-r}VR(\tau)^{r}}\right) \sim \chi^2(1), \tag{2.28}$$

where $r = \sum_{i=1}^{n} H_i(\tau)$ denotes the number of exceedances.

To check the Independence property we use the independence test from Christoffersen (1998). This test is based on a first-order Markov chain, evaluating the relation of (non-)successive VaR violations:

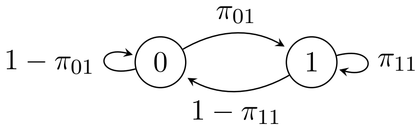


Figure 1: First order Markov Chain for Violation Transitions

where $\pi_{jk} = Pr(H_i(\tau) = k | H_{i-1}(\tau) = j)$, represents the probability of a violation given a violation or no violation in the previous observation.

The hypothesis of independence is then also tested by means of a LR test:

$$\text{LR}_{IND} = -2\ln\left(\frac{(1-VR(\tau))^{n-r}VR(\tau)^{r}}{(1-\hat{\pi}_{01})^{T_{00}}\hat{\pi}_{01}^{T_{01}}(1-\hat{\pi}_{11})^{T_{10}}\hat{\pi}_{11}^{T_{11}}}\right) \sim \chi^2(1), \tag{2.29}$$

with T_{jk} representing the number of times a j state was followed by a k state over all observations n.

In their paper, Christoffersen (1998) also combine the PoF and the Independence test to simultaneously test the Unconditional Coverage and Independence properties by creating the Conditional Coverage Independence (CCI) test, by computing:

$$\mathrm{LR}_{CCI} = \mathrm{LR}_{PoF} + \mathrm{LR}_{IND} = -2\ln\left(\frac{(1-\tau)^{n-r}\tau^{r}}{(1-\hat{\pi}_{01})^{T_{00}}\hat{\pi}_{01}^{T_{01}}(1-\hat{\pi}_{11})^{T_{10}}\hat{\pi}_{11}^{T_{11}}}\right) \sim \chi^2(2). \tag{2.30}$$

2.5.2 Robustness

Robustness is an important quality for financial market risk estimators, as it ensures the reliability and stability of predictions and resilience to human error. In theory, robust risk methods should instill greater confidence in decision-making processes and facilitate more effective risk management strategies in the financial industry. To evaluate robustness, we will consider two aspects: resilience to outliers and resilience to noise variables.

Outliers: To test the robustness of the methods to outliers, we will only focus on the simulation studies, as this gives us the opportunity to manually add outliers and see how this influences the accuracy over multiple simulations. That is, for each of the three simulation studies, we will report the $\sqrt{\mathrm{MISE}}$ (see Section 2.5.1) across various percentages of outliers within the training sets. These percentages are $\{0.1\%, 0.5\%, 1\%, 2\%, 5\%, 10\%\}$, all while maintaining a fixed quantile level $\tau = 0.99$ and a fixed dimension p. This evaluation will be conducted over 25 simulations of the simulation studies described in Sections 3.1, 3.2, and 3.3.

To introduce outliers we simulate data points from a uniform distribution using the range of observed values from the existing simulated dataset via $Y_{\text{outlier}} \sim Uniform(Y_{min}, Y_{max})$. Additionally, we generate new iterations of covariates X_{outlier} conform to the respective simulation study and link the covariates to the outliers Y_{outlier}.

Noise variables: Financial market risk analysis involves numerous possible explanatory variables we can consider. However, not all may have the significant explanatory power we initially assumed. Hence, it is pivotal for the methods to be able to cope with a certain amount of noise variables.

To assess robustness to noise variables, we will follow Gnecco et al. (2022) by assessing the performance of the methods with varying quantities of noise variables in the simulations. Specifically, will report the $\sqrt{\mathrm{MISE}}$ (see Section 2.5.1) for multiple dimensions $p = \{5, 10, 20, 40\}$ at a fixed quantile level $\tau = 0.99$. This evaluation will be conducted over 25 simulations of the simulation studies described in Sections 3.1, 3.2, and 3.3.

2.5.3 Consistency

Financial institutions and regulators desire models that remain effective and applicable for a reasonable time period, with minimal need for modifications. Consistent models reduce the frequency of building and agreeing on new models, which is a costly process. In this paper we consider a model to be consistent if it shows good predictive performance over varying DGPs.

For the simulation studies, we will assess the performance of the methods on multiple altered variations on the simulation studies described in Sections 3.1, 3.2, and 3.3 (see Appendix B for the altered DGPs). That is, we will compute the $\sqrt{\text{ISE}}$ for a fixed level of dimensionality and a fixed quantile level $\tau = 0.99$ over 50 simulations for each of the alternative DGPs (see for example Figure 9).

For the real data application, we will split the test set into five subsets and assess the accuracy of the methods for each subset. That is, we assume that for each subset the DGP changes slightly which will allow us to assess the level of consistency over the test set via the PoF- and CCI-test statistics as described in Section 2.5.1 (see for example Figure 13).

2.5.4 Importance

An effective financial risk method should accurately identify the covariates that significantly influence the outcome, enabling financial institutions to make informed and transparent decisions. Both methods use random forests, and hence for both methods, we will make use of the model-agnostic SHAP Feature Importance method from Lundberg and Lee (2017) to determine which covariates were given the most weight by the methods. This method examines the importance given to covariates by evaluating the change in the predicted outcomes when the relevant feature is shuffled randomly ceteris paribus.[3]

This component can only be assessed through our simulation studies as we cannot be sure about the true drivers behind the losses for the real data application and can therefore never say which method best identified the correct covariates.

For the simulation studies we follow Ahmed (2022) in comparing the importance ranking for the covariates that are used in the DGP. That is, we will train the methods on 50 simulations and calculate the ratio of it correctly identifying most important covariates. The methods are then compared by their respective ratios of correctly identifying the rank of the covariates (see for example Table 1).

[3] For a detailed description of SHAP see Molnar (2022).

3 Simulation Study

In this section, we will outline the three simulation methods used to comprehensively assess the quality of the two methods. As mentioned in the Introduction, we use the simulation studies from Gnecco et al. (2022) and Ahmed (2022) to assess the methods in the environment in which they were developed. In addition, we introduce a new simulation study that emulates a financial market environment.

3.1 Simulation Gnecco et al. (2022)

The simulation setup from Gnecco et al. (2022) is based on the simulation from Athey et al. (2019). Specifically, they use a setup where the scale of the response variable Y is dependent on covariates X through a step function $s(x)$. However, they differ from Athey et al. (2019) in their choice of noise distribution for the response variable, opting for the more heavy-tailed Student's t-distribution instead of a Gaussian distribution.

The setup can be described as follows: the conditional response variable is given by $Y|X = x \sim s(x)T_v$, where the predictor space is defined as $X \sim U_p$, with U_p depicting the uniform distribution on $[-1, 1]^p$. Here, T_v denotes the Student's t-distribution with v degrees of freedom, and they define the step function as

$$s(x) = 1 + \mathbb{1}\{x_1 > 0\}, \tag{3.1}$$

where x_1 is the first covariate in the p-dimensional predictor space $x \in \mathbb{R}^p$. The GPD scale parameter $\sigma(x)$ of the response variable depends only on the first covariate, indicating that all other covariates act as noise variables. In our research, we slightly adjust the scale function to:

$$s(x) = 1 + 0.8 \times \mathbb{1}\{x_1 > 0\} + 0.4 \times \mathbb{1}\{x_2 > 0\} + 0.2 \times \mathbb{1}\{x_3 > 0\}, \tag{3.2}$$

this is done to be able to better compare the performance of the methods for the 'Importance' component as described in Section 2.5.4.

Additionally, the shape parameter is independent of X and constant over the distribution at $\xi(x) = \frac{1}{v}$. Gnecco et al. (2022) use this configuration to evaluate the performance of the methods across multiple dimensions p, varying quantile levels τ, and different levels of tail heaviness $\xi(x)$.

In their appendix, Gnecco et al. (2022) expanded the simulation study by adding some extensions to the basic setup. Primarily, they made the degrees of freedom v and, therefore, the shape parameter $\xi(x)$ dependent on X through:

$$v(x) = 3[2 + \tanh(-2x_1)]. \tag{3.3}$$

Furthermore, they introduced variations in the relation between the scale parameter $\sigma(x)$ and the covariates X by employing different step functions $s(x)$ (see Gnecco et al. (2022)). However, for the purposes of our comparison, we will only use the extension of the $v(x)$ function as part of the Financial Market simulation (see Section 3.3), as estimating the reciprocal of this function, the shape parameter $\xi(x)$, is of more significance for our research.

3.2 Simulation Ahmed (2022)

The simulation study conducted by Ahmed (2022) is somewhat tailored for assessing the quality of their own method. That is, they directly define the shape parameter $\xi(x)$, the exceedance probability $g(x)$, and the threshold value u.

The setup can be described as follows: the conditional response variable $Y|X = x$ has a layered form where each ith observation is defined as

$$Y_i = \begin{cases} u\tilde{Y_i} & , \quad p_i = 1, \\ \left(\frac{F_i - F_i u^{0.1} + u^{0.1}}{u^{0.1}}\right)^{-10} & , \quad p_i = 0, \end{cases}$$

with p_i as a Bernoulli random variable, with probability $g(x_i)$, $\tilde{Y_i}$ Pareto distributed with shape parameter $\xi(x)$, $F_i \sim Uniform(0, 1)$, and u the aforementioned predefined threshold value.

$\xi(x)$ and $g(x)$ depend on the covariates X and are defined as:

$$\xi(x_i) = 0.15+0.7\times\mathbb{1}\{x_{i,3} = 2\}+0.93\times\mathbb{1}\{x_{i,2} = 2\}, g(x_i) = 0.1+0.05\times\mathbb{1}\{x_{i,2} = 3\}+0.1\times\mathbb{1}\{x_{i,1} = 2\},$$

where the covariates X are defined as $X_i = (X_{i,1}, ..., X_{i,5}), 1 \leq i \leq n$, for n observations. These five covariates are categorical variables from a multinomial distribution, which are independent of each other:

- $X_1 \sim multinom([0.7, 0.2, 0.1], n)$
- $X_2 \sim multinom([0.1, 0.5, 0.3, 0.1], n)$
- $X_3 \sim multinom([0.3, 0.4, 0.2, 0.1, 0.09], n)$
- $X_4 \sim multinom([0.7, 0.2, 0.06, 0.04], n)$
- $X_5 \sim multinom([0.8, 0.1, 0.1], n)$.

Ahmed (2022) used this setup to assess the performance with different numbers of observations n across varying quantile levels τ.

An important distinction between the Ahmed simulation study compared to the other simulation studies lies in the use of categorical variables. This feature of the Ahmed simulation study is upheld to retain the fundamental characteristics on which Ahmed (2022) assessed their methodology. In preparation for application across all methods, we applied one-hot encoding to the simulated data.

3.3 Simulation Financial Market Data

To emulate financial market data, we follow the approach of Berger and Moys (2021), which combines the eGarch(1,1) method from Nelson (1991) and the skewed t-distribution from Hansen (1994). This setup is given by:

$$\begin{aligned} r_i &= \sigma_i z_i, \\ \ln\sigma_i^2 &= \omega + \alpha\left[\left|\frac{r_{i-1}}{\sigma_{i-1}}\right| - E\left(\left|\frac{r_{i-1}}{\sigma_{i-1}}\right|\right)\right] + \gamma\frac{r_{i-1}}{\sigma_{i-1}} + \beta\ln\sigma_{i-1}^2, \\ z_i &\overset{i.i.d.}{\sim} \mathit{Skewed}\text{-t}(v,\lambda), \end{aligned} \tag{3.4}$$

where ω, α, γ, and β are regression parameters, v again denotes the degrees of freedom, and λ indicates the level of skewness.

In their simulation study, Berger and Moys (2021) estimated the parameters using the daily log-returns of the S&P500 index during the financial crisis of 2007-2008. From this, they obtained the following parameter values: $\omega = -0.2305, \alpha = 0.0264, \gamma = -0.2578, \beta = 0.975, v = 6.9003, \lambda = -0.2388$. However, to include a dependency of the degrees of freedom v on the covariates X, we borrow the dependency formula given by equation (3.3) from Gnecco et al. (2022) and make a slight adjustment similar to what we did to the scale function in equation (3.2):

$$v(x) = 3\left[2 + \tanh\left(-2\left(\frac{4}{7}x_1 + \frac{2}{7}x_2 + \frac{1}{7}x_3\right)\right)\right]. \tag{3.5}$$

We also copy the predictor space of Gnecco et al. (2022), by defining $X \sim U_p$, with U_p as the uniform distribution on $[-1, 1]^p$.

For the assessment of the accuracy, robustness, consistency, and importance components, we will directly use z_i as the return variable instead of r_i to make the comparison between the methods more transparent. We will call this simulation the Time-Invariant Financial Market simulation.

Additionally, for the assessment of the accuracy, we will also include a comparison based on the simulation of r_i, which we will call the Time-Varying Financial Market simulation.

In this Time-Varying Financial Market simulation, we will assess the performance of the methods under two variations: one variation that estimates the VaR as described in the previous sections, and one variation using an eGarch(1,1) filter on r_i before estimating the VaR. This filter involves estimating $\hat{\sigma}$ and $\hat{z}$ through an eGarch(1,1) model on the training data, after which the methods are trained on $\hat{z}$. The quantile prediction for r_i is then given by multiplying the eGarch(1,1) estimate of $\hat{\sigma}_i$ with the quantile prediction of the methods for $\hat{z}_i$. In both variations, we include the one- and two-lagged returns r_{i-1} and r_{i-2} as part of the covariate space X.

3.4 Results Simulation Study

The original objective of this research was to find which of the methods is best suited for financial market risk prediction. As discussed in Section 2.5 we will try to answer this question along four key components. That is, we will compare the performance of the methods based with regards to the prediction accuracy in Section 3.4.2, the robustness in Section 3.4.3, the consistency in Section 3.5, and the importance in Section 3.5.1.

3.4.1 Hyperparameter Tuning

To ensure the methods are evaluated under comparable conditions and to maximize their performance, we adopt the approach outlined by Ahmed (2022) in using the true quantiles for the cross-validation of the hyperparameters. However, whereas Ahmed (2022) conducted the cross-validation using the Root Mean Squared Error (RMSE), we use the MISE as the respective cross-validation error. We calculate the MISE over ten simulations for a wide range of configurations. The range of configurations and the resulting hyperparameter settings are described in Appendix C.

When comparing the performance of the methods on the S&P500 data the cross-validation schemes as described in Sections 2.3.5 and 2.4.5 are upheld (see Section 4.1.1).

3.4.2 Prediction Accuracy

The performance in terms of prediction accuracy is depicted in Figures 2, 3, 4, and 5.

In general, both ERF variations show better accuracy compared to the method from Ahmed (2022). This discrepancy in performance, is especially visible in the Gnecco simulation study, see Figure 2. Here, the performance shows a drawback of using a fixed value threshold u when the DGP has a varying scale parameter. While the ERF Pareto method also assumes a constant scale, it is able to compensate through the estimation of $\widehat{VaR}_x(\tau_0)$. The argument of the drawback is further illustrated in the Time Invariant Financial Market simulation study, see Figure 4. In that scenario, where the DGP closely resembles the DGP of the Gnecco simulation study except it has a varying shape instead of a varying scale parameter, the relative performance gap between the Ahmed algorithm and ERF methods narrows. The fixed value threshold might also explain why the largest increase in performance caused by the eGarch-filter in the Time-Varying Financial Market simulation, see Figure 5, is observed for the Ahmed algorithm. The filter estimates the scale in its stead, while the unfiltered Ahmed method occasionally overestimates the VaR by a large amount. This is similar to what we observe in the application on the S&P500 index in Section 4.1, where the eGarch filtered Ahmed method has very competitive performance while the unfiltered version drastically overestimates the VaR as $\tau \to 1$.

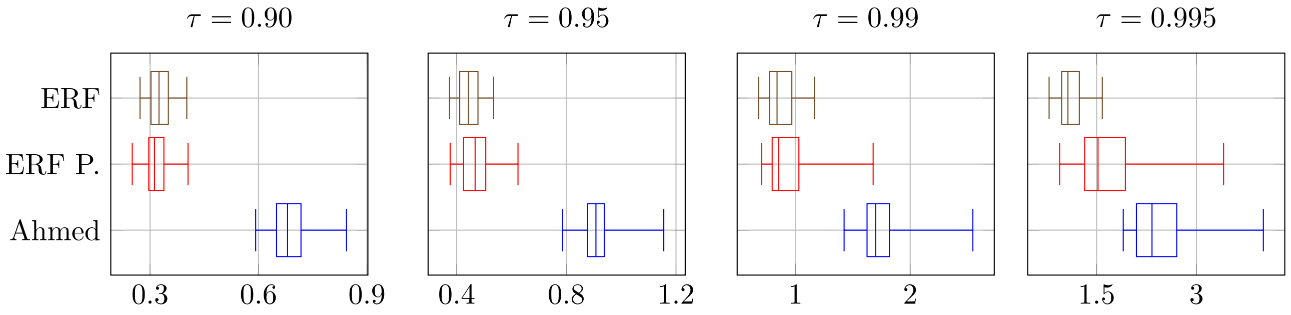


Figure 2: Boxplots of $\sqrt{\text{ISE}}$ over 50 Gnecco Simulations, with $n = 5,000$ observations and dimension $p = 10$.

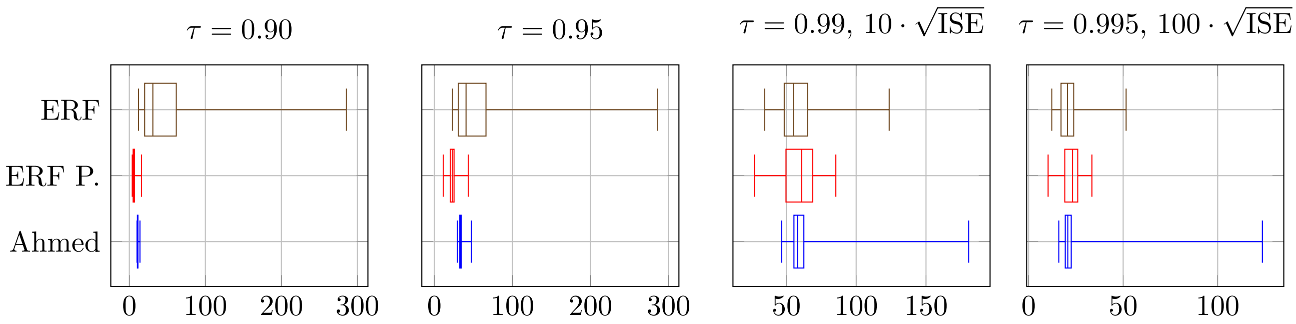


Figure 3: Boxplots of $\sqrt{\text{ISE}}$ over 50 Ahmed simulations, with $n = 10,000$ observations and dimension $p = 5$.

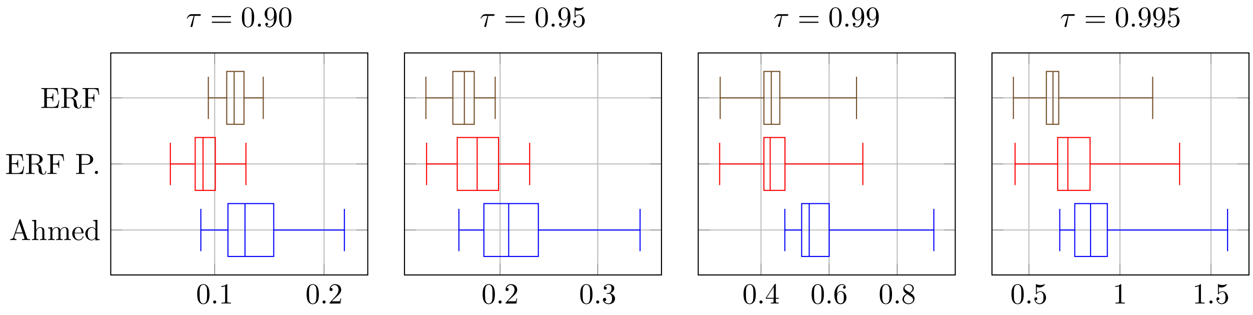


Figure 4: Boxplots of $\sqrt{\text{ISE}}$ over 50 Time-Invariant Financial Market simulations, with $n = 5,000$ observations and dimension $p = 10$.

Furthermore, we observe that all methods have instances of significant performance degradation. At closer inspection, these instances often occur where the methods associate large shape parameters with a certain part of the predictor space. This tendency leads to large overestimations of the VaR, especially for cases where $\xi(x) < 1 < \hat{\xi}(x)$. This is particularly evident in Figure 3, where the standard ERF method and the Ahmed algorithm display notably high maximum $\sqrt{\text{ISE}}$ values at $\tau = 0.90$, $\tau = 0.95$, and $\tau = 0.99$, $\tau = 0.995$, respectively. It is most pronounced in the Ahmed simulation study because the DGP has shape parameters ranging from 0.15 to 1.78 and uses only categorical variables.

Finally, we observe the comparable accuracy from the figures between the standard ERF method and its adjusted version, 'ERF Pareto', which is designed to address heavy-tailed distributions. This similarity in performance persists across all simulation studies. Even in the case of the Ahmed simulation study, as depicted in Figure 3, which is characterized by the presence of the largest shape parameters, the ERF Pareto method only shows improved accuracy for the lower quantile levels, specifically $\tau = 0.90$ and $\tau = 0.95$. However, it is at least doubtful to attribute this improved accuracy to the adjustment for heavy tails.

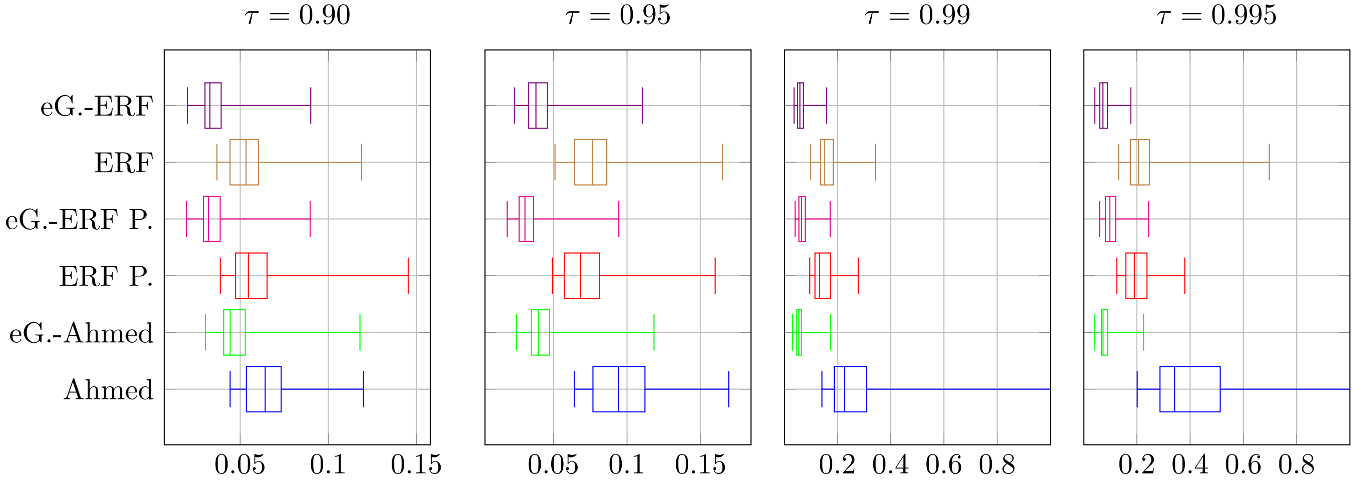


Figure 5: Boxplots of $\sqrt{\text{ISE}}$ over 50 Time-Varying Financial Market simulations, with $n = 5,000$ observations and dimension $p = 10$.
To ensure comparability, the x-axes for quantile levels $\tau = 0.99$ and $\tau = 0.995$ are constrained.

3.4.3 Robustness

The performance in terms of robustness to outliers and noise variables is represented in Figures 6, 7, and 8.

Based on the results, the methods exhibit a relatively high degree of robustness to the inclusion of noise variables. For instance, in the Gnecco simulation study, see Figure 6, there is no tangible impact on the performance of all three methods when the dimension p increases. However, in the remaining two simulation studies, see Figures 7 and 8, although the effect remains modest, the method from Ahmed (2022) seems less affected by the inclusion of noise variables compared to the ERF methods.

Conversely, the situation is different when we assess the robustness to outliers. Notably, in Figure 7, the Ahmed method exhibits a rapid decline in performance as the proportion of outliers increases. The $\sqrt{\text{MISE}}$ value (13478.920) associated with the Ahmed method at only a 0.1% outlier rate surpasses the $\sqrt{\text{MISE}}$ value (585.492) of the ERF Pareto method by over 20 times, indicating a pronounced vulnerability to outliers.

Moreover, except for the notable performance drop observed in the standard ERF method at the 1% outlier rate in the Ahmed simulation study, see Figure 7, the robustness of the standard ERF method and the adapted ERF method remain similar. This similarity is particularly evident in terms of robustness to noise variables, as indicated by the resembling trend lines across all three figures.

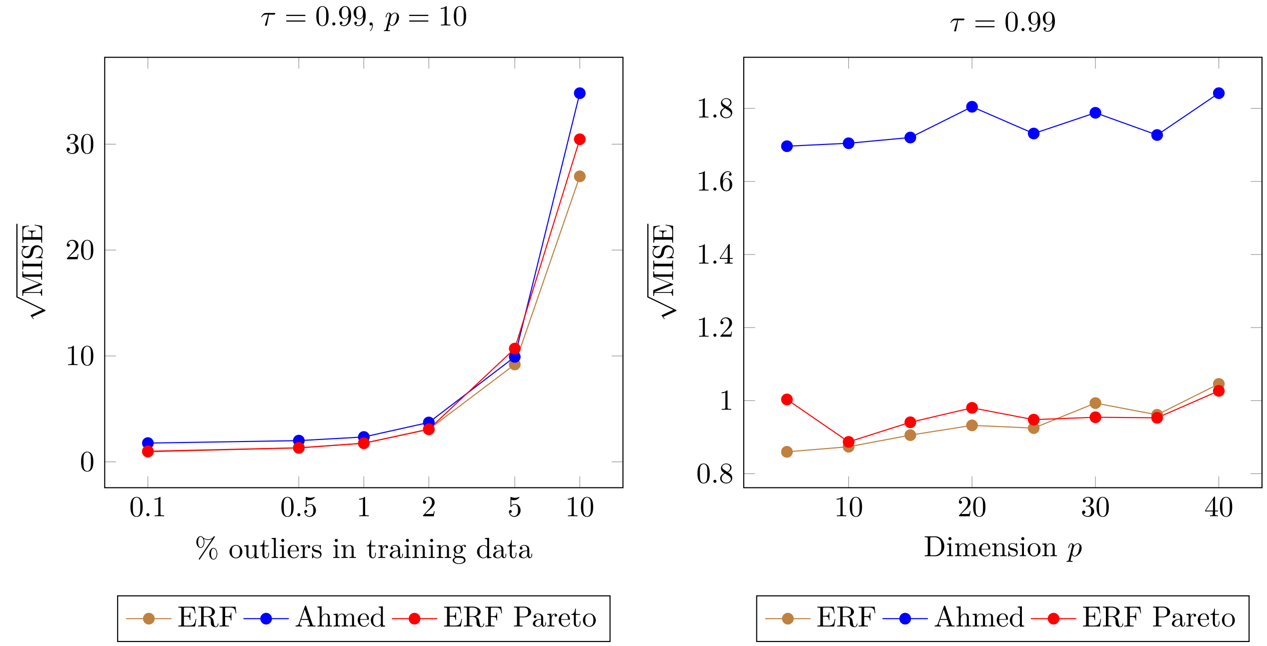


Figure 6: Robustness outliers (left) and noise variables (right) for 25 Gnecco simulations, with $n = 5,000$ observations.

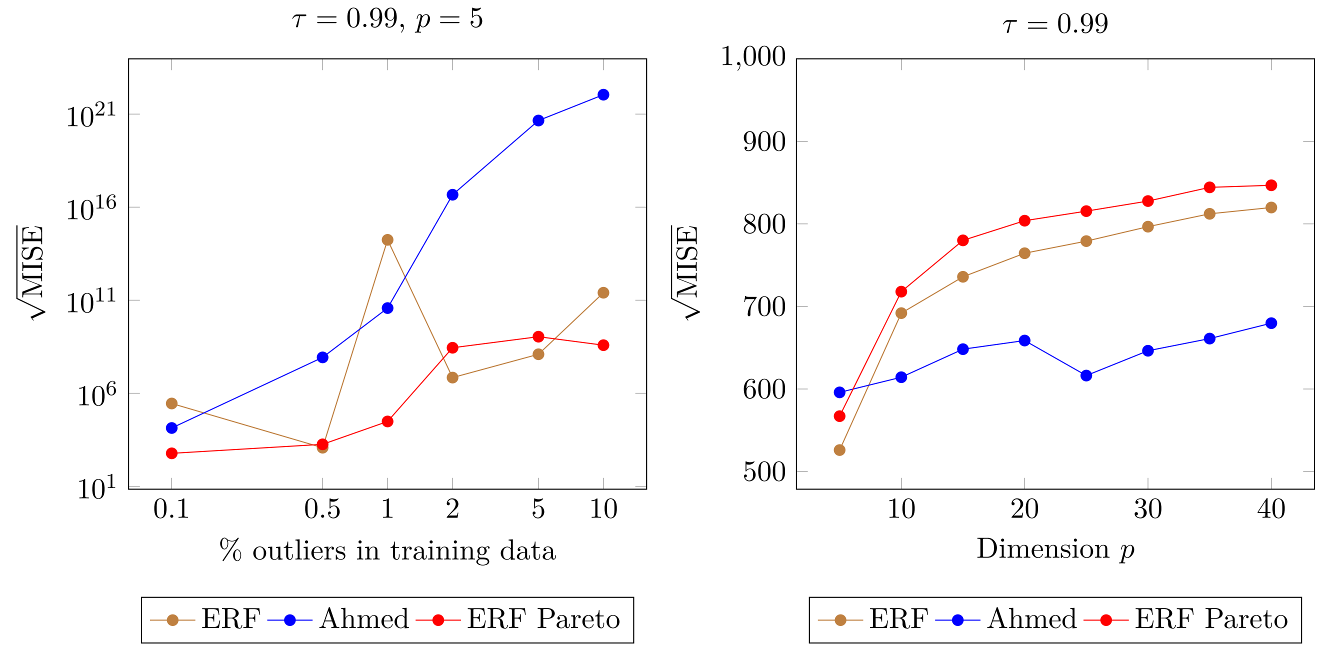


Figure 7: Robustness outliers (left) and noise variables (right) for 25 Ahmed simulations, with $n = 10,000$ observations.

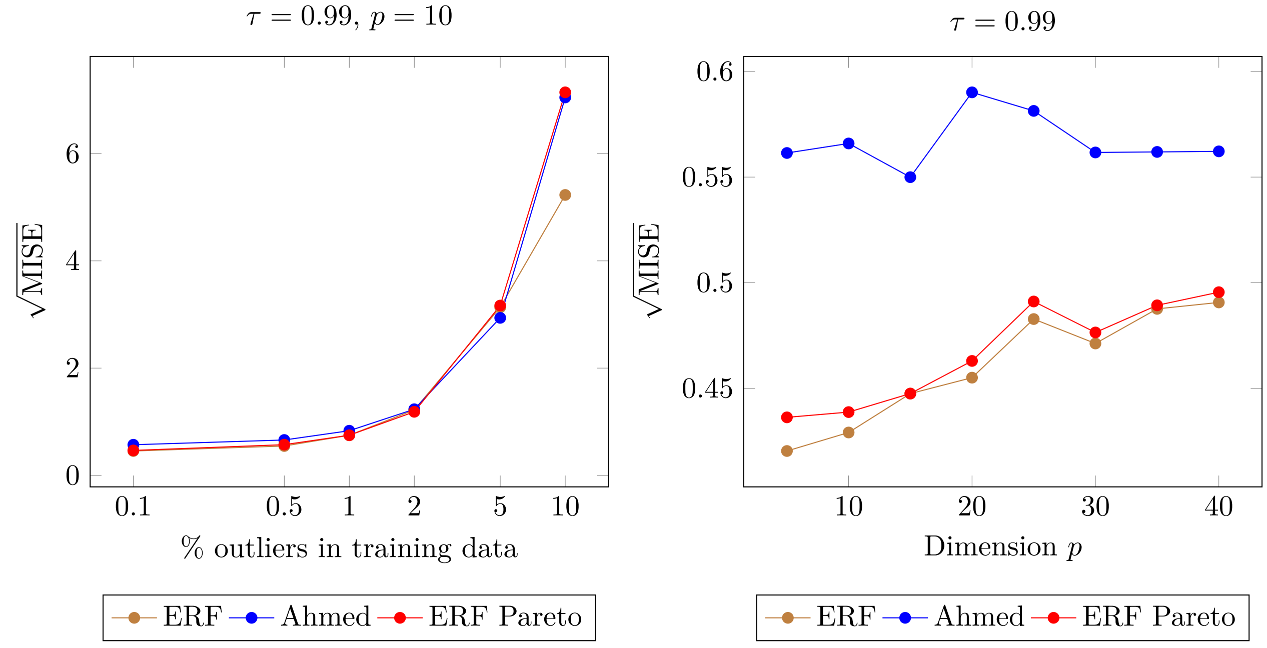


Figure 8: Robustness outliers (left) and noise variables (right) for 25 Time-Invariant Financial Market simulations, with $n = 5,000$ observations.

3.5 Consistency

The assessment of the consistency of the methods is illustrated in Figures 9, 10, and 11.

The most prominent illustration of inconsistency becomes evident in the Gnecco simulation study, specifically at $\bar{\xi} = \frac{1}{2}$ and $\bar{\xi} = \frac{5}{6}$, see Figure 9. For these particular DGPs, the Ahmed method features notably larger outliers and higher median errors in comparison to the ERF methods.

The results of the Ahmed and the Time-Invarying Financial Market simulation studies are less conclusive. In Figure 10, which concerns the Ahmed simulation study, the ERF method underperforms considerably at $\bar{\xi}(x) = 0.135$ (contrasted by the performance of the ERF Pareto method). However, at $\bar{\xi}(x) = 0.57$, the method from Ahmed (2022) once again shows negative performance outliers. A similar pattern is visible in the Time-Invariant Financial Market simulation at $\bar{\xi}(x) = \frac{1}{8}$ in Figure 11.

Overall, it is challenging to draw strong conclusions from these results, as the three simulation studies produce somewhat varied outcomes. Nevertheless, the ERF methods demonstrate more consistent results compared to the Ahmed method, characterized by smaller outliers and slightly better average performance.

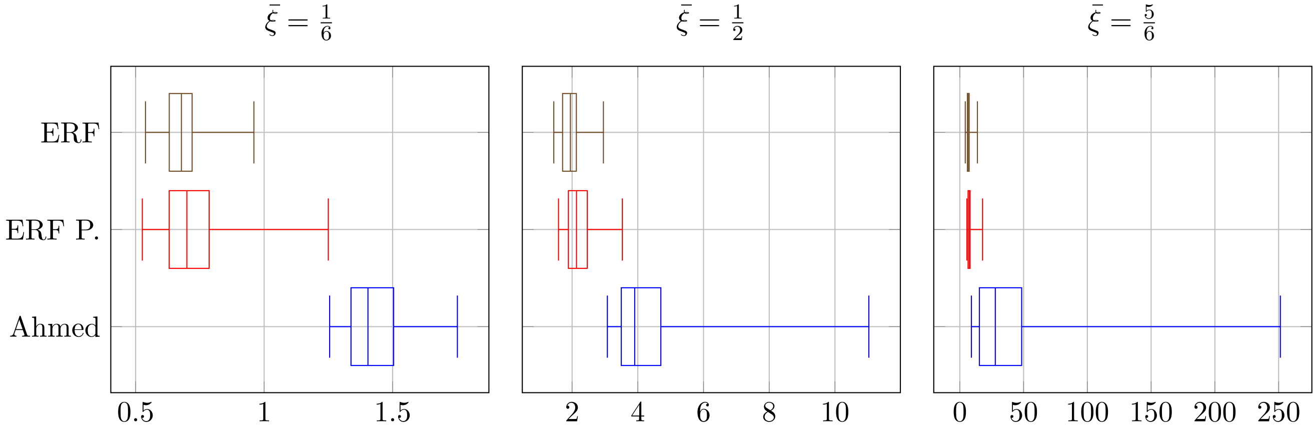


Figure 9: Boxplots of $\sqrt{\text{ISE}}$ over 50 Gnecco Simulations with $\tau = 0.99$, to represent the consistency over different levels of ξ. See Appendix B for the data generating processes.

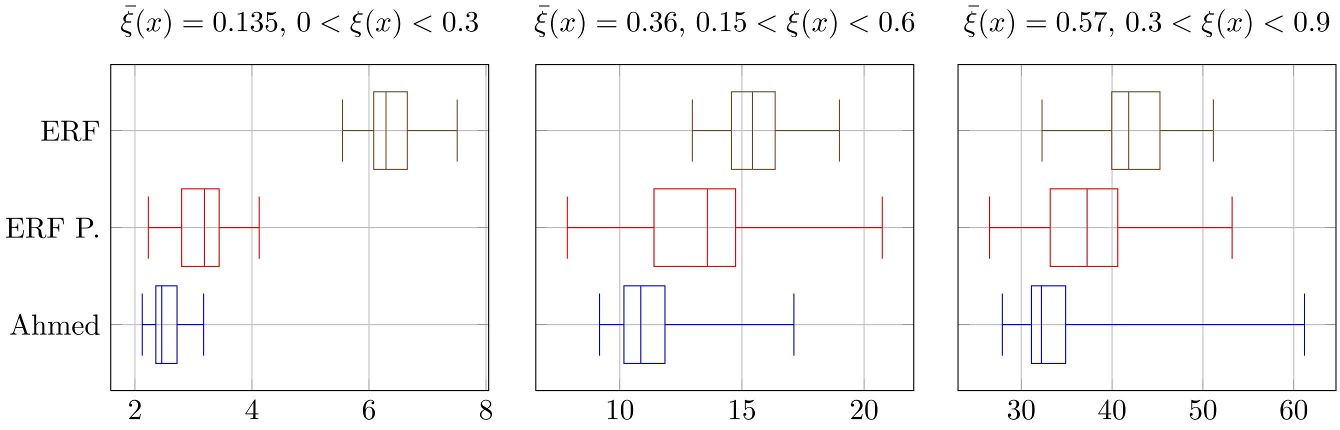


Figure 10: Boxplots of $\sqrt{\text{ISE}}$ over 50 Ahmed Simulations with $\tau = 0.99$, to represent the consistency over different levels of ξ. See Appendix B for the data generating processes.

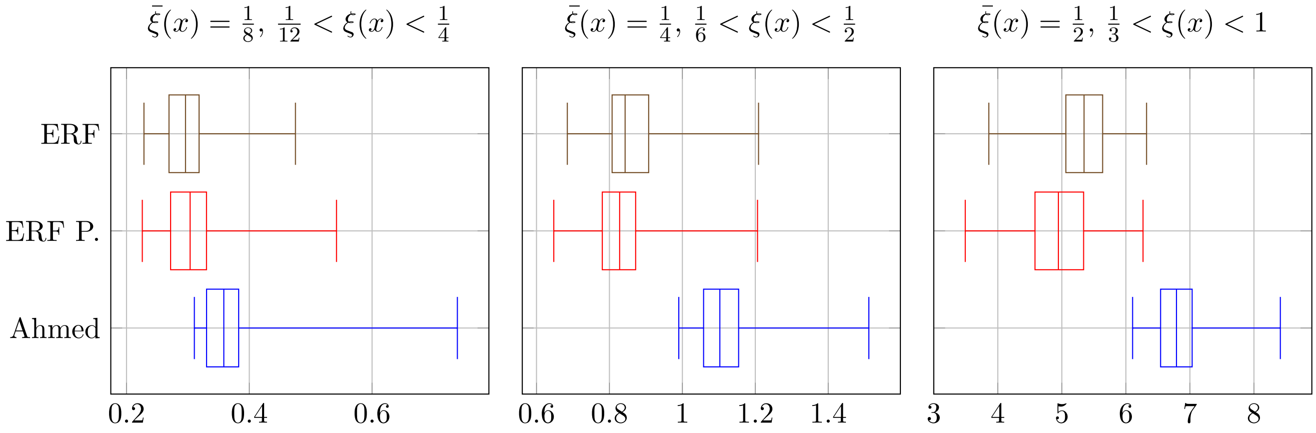


Figure 11: Boxplots of $\sqrt{\text{ISE}}$ over 50 Time-Invariant Financial Market simulations with $\tau = 0.99$, to represent the consistency over different levels of ξ. See appendix B for the data generating processes.

3.5.1 Importance

The performance of the methods with regards to the identification of the relevant covariates is presented in Tables 1, 2, and 3.

The results show that the ERF methods are better able to identify the important factors behind the DGP. This is particularly evident when examining the Gnecco and the Financial Market simulation studies, see Tables 1 and 3. In these studies, the ERF methods consistently achieve higher percentages of correctly identified rankings and also attribute higher average importance scores to the relevant covariates. This capability of the ERF methods might be a consequence of using similarity weights, see Section 2.3.3, enabling them to capture the heterogeneous relation between Y and X more effectively.

Tables 1 and 3 emphasize the challenge of identifying covariates that influence the DGP through the shape parameter rather than the scale parameter. For instance, in the Gnecco simulation study, both methods were able to correctly identify x_1 as the most important factor for all simulations, achieving a score of 100%. However, in the Financial Market simulation study, this percentage drops to 68% for the ERF method and 70% for the Pareto-adjusted ERF method, which indicates the relatively increased difficulty of identifying shape-driven covariate effects compared to scale-driven covariate effects.

Lastly, in the Ahmed simulation study, see Table 2, there is no clear distinction between the performance of the three methods. However, we do observe that the three methods struggle to distinguish between the importance of $x_2 = 2$ and $x_3 = 2$, illustrated by the closely matched average importance scores for both covariates. Furthermore, the Ahmed method seems to ignore the information stored in $x_1 = 2$, suggesting that it relies solely on $x_2 = 3$ for the estimation of $g(x)$.

Table 1: Importance ranking for the covariates used in 50 Gnecco simulations, with $n = 5,000$ observations and dimension $p = 10$.

Method	Identified First	Identified Second	Identified Third	Average importance
		Rank 1: x_1		
ERF	100%	0%	0%	0.583
ERF Pareto	100%	0%	0%	0.424
Ahmed	66%	18%	6%	0.168
		Rank 2: x_2		
ERF	0%	98%	2%	0.122
ERF Pareto	0%	92%	8%	0.124
Ahmed	6%	26%	26%	0.119
		Rank 3: x_3		
ERF	0%	2%	52%	0.053
ERF Pareto	0%	8%	46%	0.070
Ahmed	4%	8%	10%	0.095

Table 2: Importance ranking for the covariates used in 50 Ahmed simulations, with $n = 10,0000$ observations and dimension $p = 5$.

Method	Identified First	Top Two	Top Four	Correct Order*	Avg. Importance
Rank 1 $\xi(x)$-covariate: $x_2 = 2$					
ERF	54%	44%	100%	54%	0.229
ERF Pareto	40%	58%	100%	46%	0.210
Ahmed	42%	58%	100%	42%	0.309
Rank 2 $\xi(x)$-covariate: $x_3 = 2$					
ERF	46%	46%	100%	54%	0.229
ERF Pareto	54%	34%	100%	46%	0.209
Ahmed	58%	42%	100%	42%	0.312
Rank 1 $g(x)$-covariate: $x_1 = 2$					
ERF	0%	10%	92%	86%	0.139
ERF Pareto	6%	8%	94%	92%	0.136
Ahmed	0%	0%	0%	0%	0.015
Rank 2 $g(x)$-covariate: $x_2 = 3$					
ERF	0%	0%	72%	86%	0.080
ERF Pareto	0%	0%	68%	92%	0.070
Ahmed	0%	0%	100%	0%	0.097

*A covariate is considered correctly ordered when it is appropriately designated as either more or less significant than another covariate utilized within the same ξ or g function. E.g., in the case of $x_{i,2} = 2$ and $x_{i,3} = 2$, the former should be more important.

Table 3: Importance ranking for the covariates used in 50 Time-Invariant Financial Market simulations, with $n = 5,000$ observations and dimension $p = 10$.

Method	Identified First	Identified Second	Identified Third	Average importance
		RANK 1: x_1		
ERF	68%	16%	6%	0.170
ERF Pareto	70%	14%	6%	0.156
Ahmed	4%	4%	4%	0.084
		RANK 2: x_2		
ERF	16%	26%	8%	0.114
ERF Pareto	10%	26%	18%	0.113
Ahmed	18%	14%	16%	0.116
		RANK 3: x_3		
ERF	4%	14%	8%	0.095
ERF Pareto	4%	10%	12%	0.096
Ahmed	14%	8%	8%	0.103

4 Application S&P500 Index

In addition to the simulation studies, we also compare the methods in an application on actual financial market data. That is, we make use of the daily log loss returns from the Standard & Poors 500 (S&P500), obtained from the Refinitiv Datastream database. The VaR estimates are based on a combination of index-specific and various macroeconomic covariates collected from the Refinitiv Datastream, Bloomberg, FRED, and OECD databases (see Appendix A for the source and summary statistics of all the covariates).

Our dataset covers the period from January 1st, 1998, until December 31st of 2023, comprising a total of $6,587$ log loss returns. We employed a 70%/30% split, giving us a training set containing $4,611$ observations and a test set with $1,976$ observations.

Given that stock data often exhibits serial dependence, as was pointed out in (Baltussen, van Bekkum, & Da, 2019), it is important to address this issue to prevent bias in our models. As illustrated by McNeil et al. (2015), the existence of serial dependence can have a large negative impact on the performance of both Hill-based and GPD-based tail estimates and could therefore affect both methods. Hence, to mitigate this issue we will make use of the same eGarch-filter we used for in the Time-Varying Financial Market simulation. The filter is based on the Garch-filter developed by McNeil and Frey (2000) which was previously implemented on the S&P500 index by Paul and Sharma (2021).

4.1 Results Application S&P 500

Similar to the results for the simulation studies (see Section 3.4), we split the results into subsections containing results on the prediction accuracy (see Section 4.1.3) and consistency (see Section 4.1.4). Additionally, we start with a section on the observed VaR estimates (see Section 4.1.2), to illustrate the differences between the methods in this application. We do not include sections on robustness and importance for the S&P500 application, because we cannot objectively compare the performance of the methods for the components with an empirical data set.

4.1.1 Hyperparameter Tuning

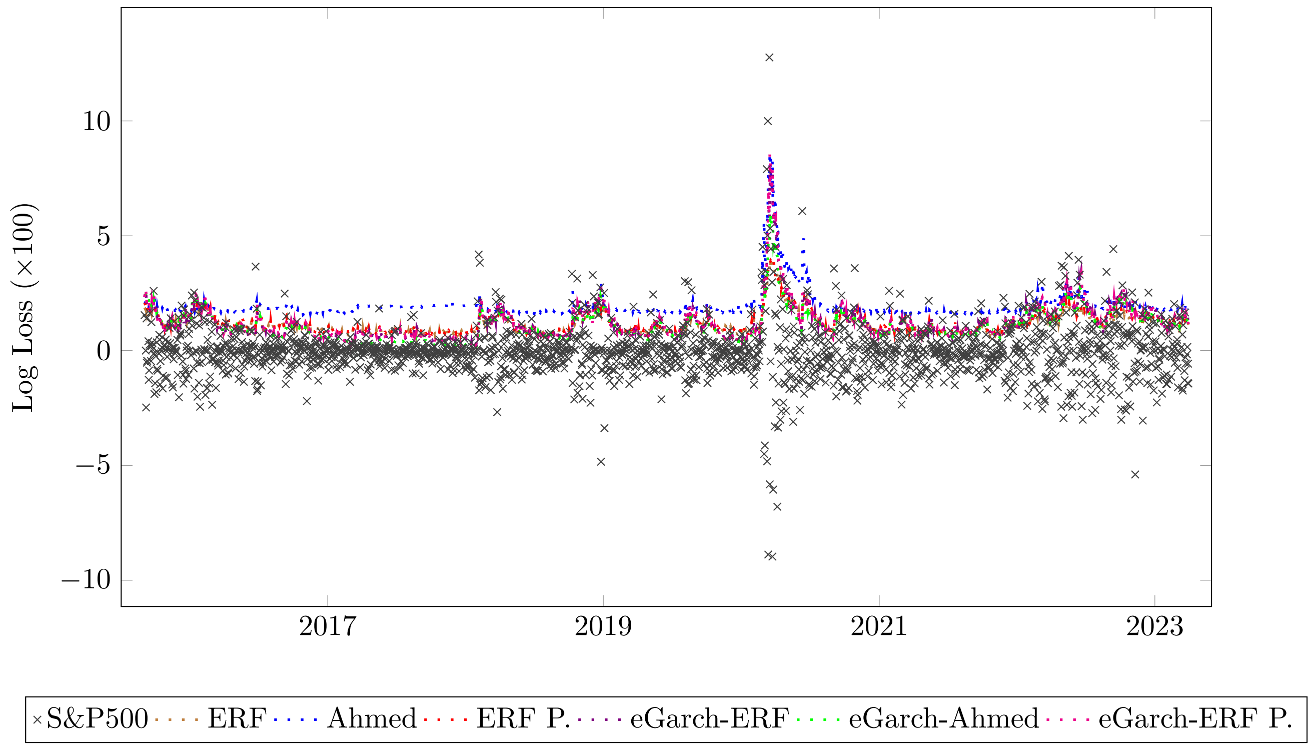
For the application of the S&P500 data, we employ the cross-validation schemes as described in Sections 2.3.5 and 2.4.5 to find the hyperparameter settings for the methods. Notably, the cross-validation process for the Ahmed Algorithm was based around the $\tau = 0.995$ quantile level, whereas the ERF cross-validation scheme is independent of the quantile level. Details regarding the range of configurations and the resulting hyperparameter settings are described in Appendix B.

4.1.2 VaR Estimates S&P500

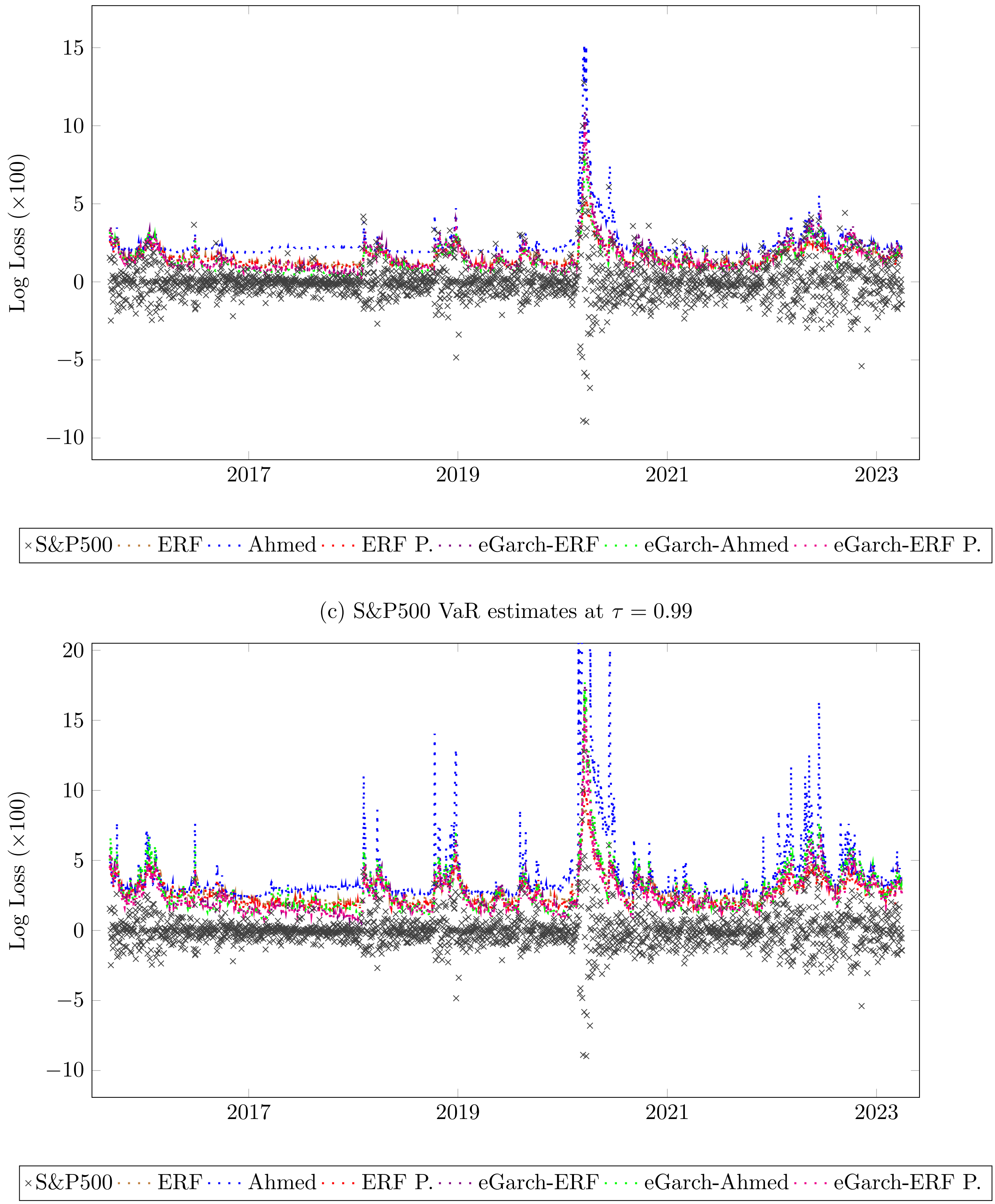
The observed VaR estimates are depicted in Figure 12.

Figure 12 provides valuable insights into the relationship between the VaR estimates generated by the six estimators and the empirical Log Losses associated the S&P500 Index. From this figure, we observe that, while most methods follow similar estimation trend lines, the Ahmed method exhibits a much flatter estimation line. This flat line is particularly evident in subfigures (a) and (b) of Figure 12, and in subfigures (c) and (d) we observe much higher peaks from the method. The flatness of the Ahmed estimation line can be attributed to the fact that the threshold ($u = 2$) is probably too high for these lower quantile levels. This once again shows the drawback of using a fixed value threshold, especially in scenarios where the scale parameter varies over time. The absence of these problems in the filtered Ahmed method reaffirms what we saw in the Time-Varying Financial Market simulation study, see Section 3.4.2, emphasizing that the Ahmed method benefits the most from filtering the time series data with eGarch(1,1).

Overall, the results in Figure 12 do not offer a clear distinction in the predictive performance of the methods. However, we do observe the notable underperformance of the Ahmed method. Additionally, noteworthy observations are the relatively high VaR estimates of the eGarch-Ahmed method at the upper quantile levels of $\tau = 0.99$ and $\tau = 0.995$ compared to other estimators (except the unfiltered Ahmed method), and, conversely, the relatively low VaR estimates of the eGarch-ERF Pareto method. These observations may indicate either positive or negative standout performances.

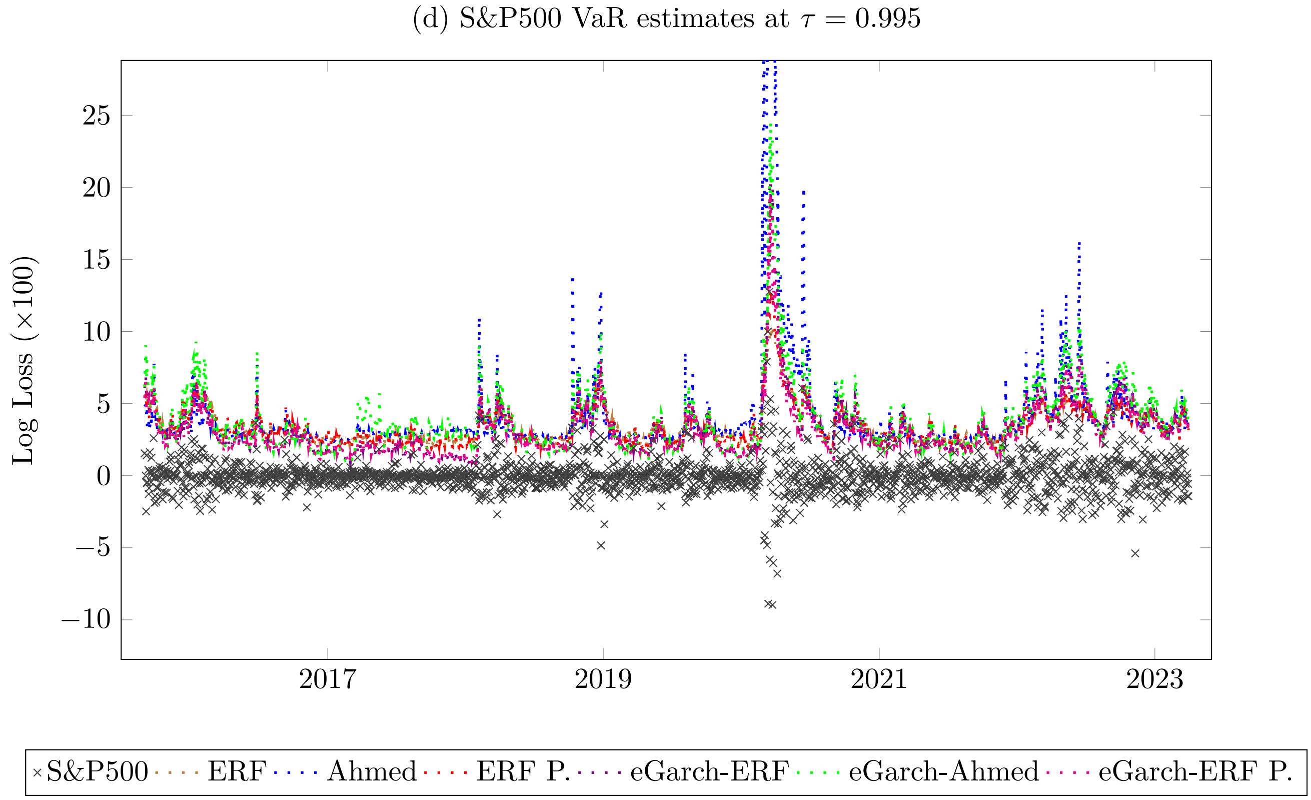


(a) S&P500 VaR estimates at $\tau = 0.9$



(b) S&P500 VaR estimates at $\tau = 0.95$

(c) S&P500 VaR estimates at $\tau = 0.99$



(d) S&P500 VaR estimates at $\tau = 0.995$

Figure 12: VaR estimates of the filtered and unfiltered ERF and Ahmed methods plotted against the actual Log Loss of the S&P500 Index.
To ensure comparability, the y-axes for quantile levels $\tau = 0.99$ and $\tau = 0.995$ are constrained.

4.1.3 Prediction Accuracy S&P500

The performance of the methods in terms of prediction accuracy is presented in Table 4.

Before discussing the accuracy results for the S&P500 data set, we have to acknowledge the fact that we cannot draw too strong conclusions from these results, since we only have one historical outcome. Especially since we saw from the simulation studies that the methods had certain instances in which their performance deteriorated significantly. Nevertheless, Table 4 yields interesting results that align with what we saw in the simulation studies.

Firstly, we note that the ERF methods, when unfiltered, demonstrate the most promising performance, which is consistent with our observations in Section 3.4.2. Furthermore, there is no clear indication which of the ERF configurations is the most accurate; for instance, we observe higher P-values for the standard ERF method at $\tau = 0.99$ and higher P-values for the ERF Pareto method at $\tau = 0.995$.

Additionally, consistent with our findings from the Time-Varying Financial Market simulation study, the eGarch filtering significantly enhances the performance of the Ahmed method. The improvement is so substantial, that the performance based on the Pof test and the CCI test cannot be distinguished from the unfiltered ERF methods. However, contrary to the results from the Time-Varying Financial Market simulation study, filtering has a negative impact on the ERF methods. This divergence from the simulation studies may be attributed to differences in the cross-validation scheme; the ERF methods employ filtered data in the calculation of the negative log-likelihood contribution, while the eGarch-Ahmed method utilizes the Brier score and the PoF Test Statistic for cross-validation.

Lastly, as discussed in the previous section (see Section 4.1.2), we alluded to the possibility of distinctly positive or negative relative performance for the eGarch-ERF Pareto and eGarch-Ahmed methods. Table 4 shows that the eGarch-ERF Pareto tends to underestimate the VaR, resulting in low P-values. Conversely, the eGarch-Ahmed does have relatively good predictive performance, with P-values well within the commonly used acceptance regions. However, it is worth noting that the eGarch-Ahmed method also exhibits significantly higher average VaR estimates than the ERF Pareto methods, while suffering more violations than the ERF Pareto method at quantile level $\tau = 0.995$. This suggests a propensity to overestimate the VaR, which could be seen as a drawback for financial institutions (overestimating risk leads to unnecessarily high capital reserves), or a positive feature for regulatory institutions (overestimating risk leads to more conservative capital reserves).

Table 4: Accuracy VaR predictions S&P500

$\tau = 0.90$

		PoF Test		CCI Test	
Method	Average VaR Estimate	#Violations (%)	P-value	Test Statistic	P-value
ERF	1.163	177 (8.957)	0.116	8.230	0.016
ERF Pareto	1.223	162 (8.198)	0.006	11.290	0.004
Ahmed	1.952	78 (3.947)	0.000	106.220	0.000
eGarch-ERF	1.222	182 (9.211)	0.236	3.429	0.180
eGarch-ERF Pareto	1.245	174 (8.806)	0.071	6.585	0.037
eGarch-Ahmed	1.080	220 (11.134)	0.098	7.241	0.027

$\tau = 0.95$

		PoF Test		CCI Test	
Method	Average VaR Estimate	#Violations (%)	P-value	Test Statistic	P-value
ERF	1.651	93 (4.706)	0.546	3.169	0.205
ERF Pareto	1.550	105 (5.314)	0.526	1.555	0.460
Ahmed	2.407	41 (2.074)	0.000	45.308	0.000
eGarch-ERF	1.706	79 (3.998)	0.034	6.787	0.034
eGarch-ERF Pareto	1.530	116 (5.870)	0.084	4.608	0.100
eGarch-Ahmed	1.431	139 (7.034)	0.000	16.063	0.000

$\tau = 0.99$

		PoF Test		CCI Test	
Method	Average VaR Estimate	#Violations (%)	P-value	Test Statistic	P-value
ERF	2.893	20 (1.012)	0.957	0.432	0.806
ERF Pareto	2.693	26 (1.316)	0.178	2.531	0.282
Ahmed	4.105	8 (0.405)	0.003	9.196	0.010
eGarch-ERF	2.789	27 (1.366)	0.121	6.135	0.047
eGarch-ERF Pareto	2.471	35 (1.777)	0.002	11.730	0.003
eGarch-Ahmed	2.791	24 (1.215)	0.354	1.975	0.372

$\tau = 0.995$

		PoF Test		CCI Test	
Method	Average VaR Estimate	#Violations (%)	P-value	Test Statistic	P-value
ERF	3.479	12 (0.607)	0.513	0.587	0.746
ERF Pareto	3.418	9 (0.455)	0.776	0.173	0.917
Ahmed	5.303	4 (0.202)	0.033	4.564	0.102

eGarch-ERF	3.238	17 (0.860)	0.040	6.480	0.039
eGarch-ERF Pareto	3.038	21 (1.063)	0.002	15.044	0.001
eGarch-Ahmed	3.757	10 (0.506)	0.970	0.113	0.945

Note: this table presents the results for the PoF Test and the CCI Test as described in Section 2.5.1.

4.1.4 Consistency

The performance of the methods with regards to consistency is presented in Figures 13 and 14.

Again, we have to note that the following results are based on a limited amount of data and should therefore be interpreted carefully.

In both figures, we observe that the non-filtered methods exhibit shorter performance spans across the five subsets compared to the filtered methods. This distinction is most apparent in Figure 14, where the non-filtered methods have the same P-value for at least two or more subsamples. Furthermore, the fact that the filtered methods have both the highest and the lowest P-values for both quantiles suggests that the effectiveness of the eGarch filter varies over time.

The results in Figure 13 show that the Ahmed method displays the most consistency, as it has the shortest performance span. However, in the previous sections on the S&P500 results we saw the consistent overestimation by the Ahmed method, which complicates this result. That is, when a method consistently overestimates the VaR, a comparison of the consistency based on the violations will show high consistency. Additionally, Figure 14 reveals a wider performance span for the Ahmed method, with the ERF methods, especially ERF Pareto, exhibitihng greater consistency.

While we cannot convincingly conclude which method is most consistent from these results, they do not contradict the results on consistency from the simulation studies that suggested the Ahmed method may be less consistent compared to the ERF method (see Section 3.5).

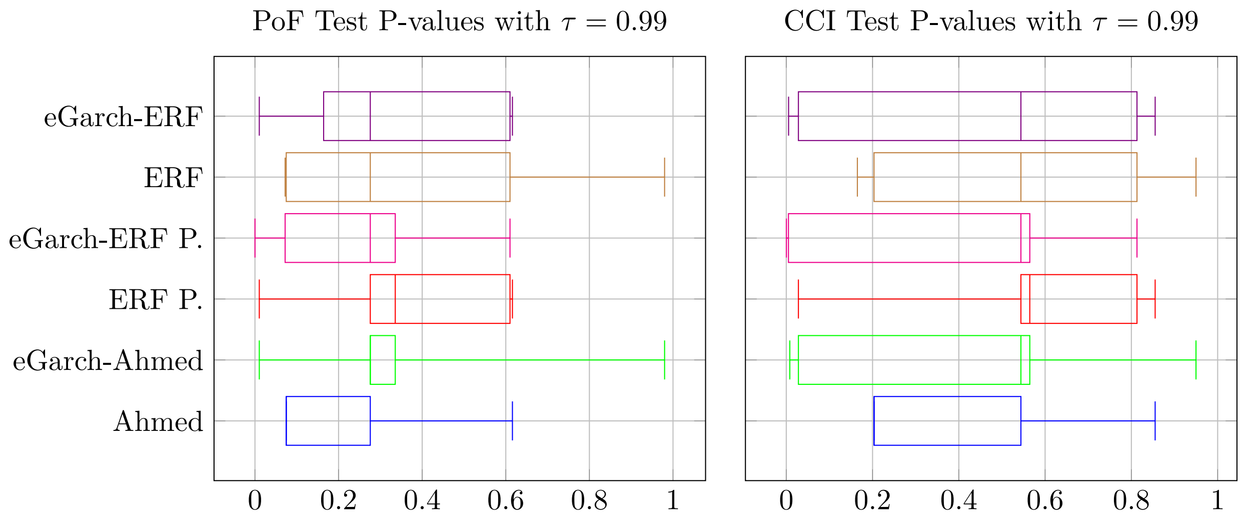


Figure 13: Boxplots of the P-values for the PoF Test (left) and the CCI Test (right) at quantile level $\tau = 0.99$, to represent the consistency over five subsets of the test set.

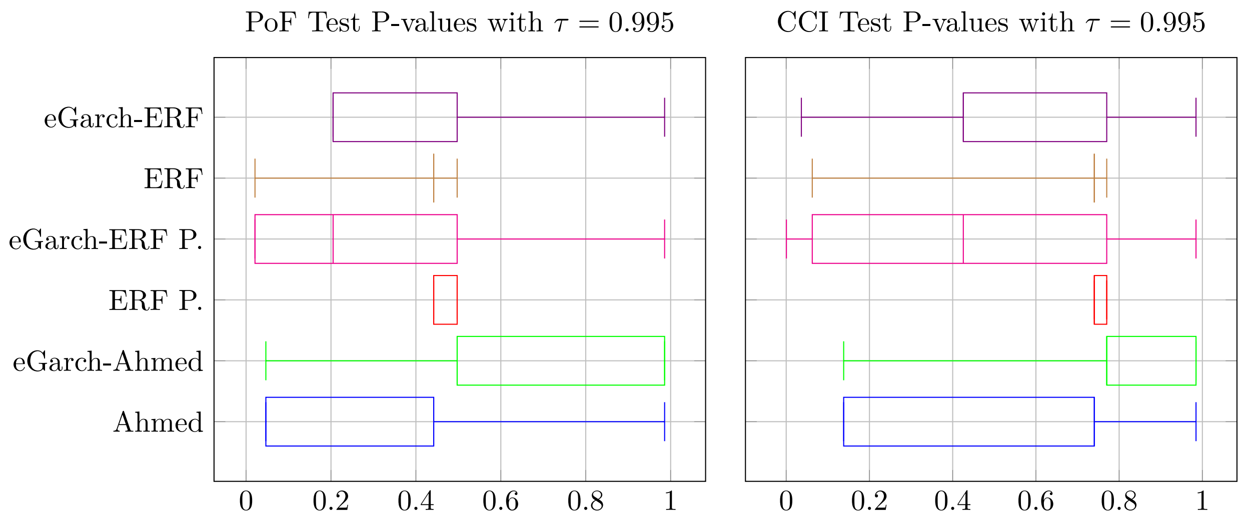


Figure 14: Boxplots of the P-values for the PoF Test (left) and the CCI Test (right) at quantile level $\tau = 0.995$, to represent the consistency over five subsets of the test set.

5 Conclusion

In our research, we conducted a comprehensive examination of two new Random Forest tail risk estimation methods, aiming to address the question of which of the methods is better suited to financial market risk prediction. We aimed to answer this question by evaluating four key quality components for financial market risk models for the methods over three simulation studies and an application involving data from the S&P500 Index.

Our findings from the simulation studies suggested that the ERF methodology (both the standard ERF and the Pareto adjusted ERF) from Gnecco et al. (2022) in general showed better predictive performance, more consistent results at higher shape parameters, and could also best identify the most important covariates driving the data generating process. When looking at the robustness, the methods showed similar performance, with the approach from Ahmed (2022) displaying greater sensitivity to outliers and the ERF method showing more susceptibility to noise variables. In summary, we conclude that the results from the simulation studies suggest that the ERF methodology is best suited for financial market risk prediction. Nevertheless, it is worth noting that the eGarch-filtered Ahmed method showed promising predictive accuracy in the Time-Varying Financial Market simulation study.

Our application of the methods on the S&P500 Index confirmed many of our findings from the results in the simulation studies. In particular, the application illustrated the drawback of using a fixed value threshold (u), instead of a fixed quantile level threshold (τ_0) and exposed the problems of the method from Ahmed (2022) with data generating processes with varying scale. Notably, we also confirmed the promising results of the eGarch-filtered Ahmed method, which seemed to reduce the problem of the time-varying scale. Conversely, in contrast with the Time-Varying Financial Market simulation study, filtering had a negative impact on the ERF methods, which we suggested might be attributable to cross-validation problem. Our investigation into consistency, though constrained by a relatively small dataset, suggested that filtered models tend to exhibit lower consistency compared to their non-filtered counterparts. Overall, the stock market application confirmed our conclusion from the simulation studies that the ERF methodology from Gnecco et al. (2022) is more suited to financial market risk prediction.

In light of these findings, our recommendation for practical implementation would be to use a combination of ERF, ERF Pareto, and the eGarch-Ahmed method for financial market risk prediction. This diversified approach would account for the fact that all methods showed moments where their performance deteriorated. A financial regulator could, for instance, oblige important financial institutions to base their capital reserves on the most conservative risk estimate the three methods produce.

Lastly, we would like to acknowledge that there are some limitations to our research for which further research might be needed. Firstly, our data application suffered from a limited amount of data and focused solely on single index. Future research endeavors should extend this evaluation to diverse continents, economies, and indices to validate our conclusions across varied contexts. Furthermore, the substantial performance improvement observed by the eGarch filtering for the Ahmed method implies the potential for fine-tuning the methodologies. For instance, based on the results of the ‘importance’ component, see Section 3.5.1), we recommend exploring the integration of similarity weights from the GRF method (Athey et al., 2019) into the Ahmed methodology. This might improve its ability to understand the complex relation between X and Y. Lastly, we based our analysis on a subset of components and statistical tests; it remains possible that alternative test or components could have yielded different conclusions.

References

Abad, P., Benito, S., & López, C. (2014). A comprehensive review of value at risk methodologies. *The Spanish Review of Financial Economics*, *12*(1), 15–32.

Ahmed, H. (2022). Extreme value statistics using related variables.

Athey, S., Tibshirani, J., & Wager, S. (2019). Generalized random forests.

Balkema, A. A., & De Haan, L. (1974). Residual life time at great age. *The Annals of probability*, *2*(5), 792–804.

Baltussen, G., van Bekkum, S., & Da, Z. (2019). Indexing and stock market serial dependence around the world. *Journal of Financial Economics*, *132*(1), 26–48.

Berger, T., & Moys, G. (2021). Value-at-risk backtesting: Beyond the empirical failure rate. *Expert Systems with Applications*, *177*, 114893.

Berkowitz, J., Christoffersen, P., & Pelletier, D. (2011). Evaluating value-at-risk models with desk-level data. *Management Science*, *57*(12), 2213–2227.

Bradley, B. O., & Taqqu, M. S. (2003). Financial risk and heavy tails. In *Handbook of heavy tailed distributions in finance* (pp. 35–103). Elsevier.

Breiman, L. (2001). Random forests. *Machine learning*, *45*, 5–32.

Brier, G. W. (1950). Verification of forecasts expressed in terms of probability. *Monthly weather review*, *78*(1), 1–3.

Bücher, A., & Segers, J. (2017). On the maximum likelihood estimator for the generalized extreme-value distribution. *Extremes*, *20*, 839–872.

Campbell, S. D. (2005). A review of backtesting and backtesting procedures.

Chavez-Demoulin, V., Embrechts, P., & Hofert, M. (2016). An extreme value approach for modeling operational risk losses depending on covariates. *Journal of Risk and Insurance*, *83*(3), 735–776.

Chernozhukov, V. (2005). Extremal quantile regression.

Christoffersen, P. F. (1998). Evaluating interval forecasts. *International economic review*, 841–862.

de Haan, L., & Ferreira, A. (2006). *Extreme value theory: an introduction* (Vol. 3). Springer.

Echaust, K., & Just, M. (2020). Value at risk estimation using the garch-evt approach with optimal tail selection. *Mathematics*, *8*(1), 114.

Embrechts, P., Klüppelberg, C., & Mikosch, T. (2013). *Modelling extremal events: for insurance and finance* (Vol. 33). Springer Science & Business Media.

Farkas, S., Lopez, O., & Thomas, M. (2021). Cyber claim analysis using generalized pareto regression trees with applications to insurance. *Insurance: Mathematics and Economics*, *98*, 92–105.

Gencay, R., & Selçuk, F. (2004). Extreme value theory and value-at-risk: Relative performance in emerging markets. *International Journal of forecasting*, *20*(2), 287–303.

Gilli, M., & Këllezi, E. (2006). An application of extreme value theory for measuring financial risk. *Computational Economics*, *27*, 207–228.

Gnecco, N., Terefe, E. M., & Engelke, S. (2022). Extremal random forests. *arXiv preprint arXiv:2201.12865*.

Gordon, A., Breiman, L., Friedman, J., Olshen, R., & Stone, C. J. (1984). Classification and regression trees. *Biometrics*, *40*(3), 874.

Hansen, B. E. (1994). Autoregressive conditional density estimation. *International Economic Review*, 705–730.

Koenker, R., & Bassett, G. (1978). Regression quantiles. *Econometrica: journal of the Econometric Society*, 33–50.

Kupiec, P. (1995). Techniques for verifying the accuracy of risk measurement models. *The J. of Derivatives*, *3*(2).

Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. In *Proceedings of the 31st international conference on neural information processing systems* (pp. 4768–4777).

Mandelbrot, B. (1963). e variation of certain speculative prices, e journal of business, 36 (4), 394-419. *DOI: http://dx. doi. org/10.1086/294632*.

McNeil, A. J., & Frey, R. (2000). Estimation of tail-related risk measures for heteroscedastic financial time series: an extreme value approach. *Journal of empirical finance*, *7*(3-4), 271–300.

McNeil, A. J., Frey, R., & Embrechts, P. (2015). *Quantitative risk management: concepts, techniques and tools-revised edition*. Princeton university press.

Meinshausen, N., & Ridgeway, G. (2006). Quantile regression forests. *Journal of machine learning research*, *7*(6).

Molnar, C. (2022). A guide for making black box models explainable. *URL: https://christophm. github. io/interpretable-ml-book*.

Nelson, D. B. (1991). Conditional heteroskedasticity in asset returns: A new approach. *Econometrica: Journal of the Econometric Society*, 347–370.

Niculescu-Mizil, A., & Caruana, R. (2005). Predicting good probabilities with supervised learning. , 625–632.

Paul, S., & Sharma, P. (2021). Forecasting gains by using extreme value theory with realised garch filter. *IIMB Management Review*, *33*(1), 64–70.

Pickands III, J. (1975). Statistical inference using extreme order statistics. *the Annals of Statistics*, 119–131.

Platt, J., et al. (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, *10*(3), 61–74.

Robnik-Šikonja, M., & Bohanec, M. (2018). Perturbation-based explanations of prediction models. In *Human and machine learning* (pp. 159–175). Springer.

Staudt, Y., & Wagner, J. (2021). Assessing the performance of random forests for modeling claim severity in collision car insurance. *Risks*, *9*(3), 53.

Taleb, N. N. (2007). *The black swan: The impact of the highly improbable.* Random House.

The Economist. (1999). The price of uncertainty. *The Economist*, *June 12th 1999*, 81–82.

Velthoen, J., Dombry, C., Cai, J.-J., & Engelke, S. (2021). Gradient boosting for extreme quantile regression. *arXiv preprint arXiv:2103.00808*.

Wang, H. J., & Li, D. (2013). Estimation of extreme conditional quantiles through power transformation. *Journal of the American Statistical Association*, *108*(503), 1062–1074.

Weissman, I. (1978). Estimation of parameters and large quantiles based on the k largest observations. *Journal of the American Statistical Association*, *73*(364), 812–815.

A Data Description and Summary Statistics

Table 5: Data Description and Summary Statistics of Data Used for S&P500 Application.

Name	Description	Mean	Min.	Max.	#NAs	Imp. Method	Used
	S&P500 Specific Sourced Variables (Source: Datastream)						
Price Closing	Closing price (D)	1840.8	676.5	4796.6	0	NA	Y
Price Opening	Opening price (D)	1840.5	679.3	4804.5	234	Kalman	Y
Price High	Highest obs. price (D)	1851.6	695.3	4818.6	234	Kalman	Y
Price Low	Lowest obs. price (D)	1828.5	665.8	4780.0	234	Kalman	Y
Dividend Yield	Associated Div. Yield (D)	1.831	1.040	3.610	0	NA	Y
	S&P500 Specific Sourced Variables (Source: Bloomberg)						
Transaction Volume	Volume of Stocks Traded (D)	9.238×10^8	1.134×10^6	2.953×10^9	233	Kalman	Y
Volatility 10 Day	Implied 10-day volatility (D)	16.75	2.34	127.32	233	Kalman	Y
Volatility 30 Day	Implied 30-Day volatility (D)	17.26	3.41	87.99	234	Kalman	Y
Volatility 90 Day	Implied 90-Day volatility (D)	17.88	5.50	65.05	234	Kalman	Y
Volatility 180 Day	Implied 180-Day volatility (D)	18.27	6.74	53.03	234	Kalman	Y
	S&P500 Specific Transformed Variables						
Log Loss Return	Neg. Log Ret. via Price Closing (D)	-0.022	-10.957	12.765	0	NA	NA
L.Loss Return 1D Prior	Lagged Log Loss Return (D)	-0.022	-10.957	12.765	0	NA	Y

L.Loss Return 2D Prior	Two-day lagged Log Loss Return (D)	-0.022	-10.957	12.765	0	NA	Y
Price 5D Avg.	Five-day moving average (D)	1839.9	690.3	4785.6	0	NA	Y
Price 10D Avg.	Ten-day moving average (D)	1838.7	707.9	4765.8	0	NA	Y
Bias	(Price Closing - Price 5D Avg.) /Price 5D Avg. (D)	0.000	-0.103	0.007	0	NA	Y
DMA	Price 5D Avg. - Price 10D Avg.	1.151	-190.686	112.817	0	NA	Y
CDP/Lagged HLCC4	(Price High Prev. D. + Price Low Prev. D. + 2 × Price Closing Prev. Day)/4 (D)	1840.0	680.3	4795.0	0	NA	Y
AR	(Price High - Price Opening) /(Price Opening - Price Low)×100 (D)	98.66	50.00	150.00	0	NA	Y
BR	(Price High - Price Closing) /(Price Closing - Price Low)×100 (D)	138.9	50.0	300.0	0	NA	Y
Percent Change	(Price High - Price Low) /(Price Low) ×100 (D)	1.352	0.146	11.521	0	NA	Y
Overnight Spread	(Price Opening - Price Closing Prev. Day) (D)	0.150	202.430	111.320	0	NA	Y
Macroeconomic Sourced Variables (Source: FRED)							
Oil Price	Brent Crude Oil Price (D)	61.60	9.10	143.95	180	Kalman	N
Inflation Exp. 5 Year	5-Y. Forward Inflation Expectation Rate (D)	1.785	-2.240	3.590	1305	Kalman	Y
Inflation Exp. 10 Year	10-Y. Forward Inflation Expectation Rate (D)	1.989	0.040	3.020	1305	Kalman	Y
FFR Percent	Federal Funds Rate % (D)	1.954	0.040	7.060	0	NA	Y

Money Supply ×B.	Money Supply M1 (W)	4017	1058	21016	37	Mov. Avg.	N
FSI	Financial Stress Index (W)	0.047	-1.162	9.315	35	Kalman	Y
Inflation	Sticky Price CPI less Food and Energy (M)	2.512	2.512	6.617	0	NA	Y
CPI Expectation	Univ. of Michigan: Inflation Expectation (M)	3.046	0.400	5.400	0	NA	Y
Unemployment	Unempl. as a percentage of the labor force (M)	5.716	3.400	14.700	0	NA	Y
Real GDP Per Capita	Real GDP per capita (Q)	51721	43062	60611	0	NA	N
Public Debt To GDP	Total Public Debt as % of GDP (Q)	84.67	54.03	134.84	0	NA	Y
Private Debt×B.	Household and Non-Profit Org. Debt (Q)	12800	5771	19159	0	NA	N
GDP×B.	Gross Domestic Product (Q)	15873	8866	26530	0	NA	N
FDI	Foreign Direct Investment (Q)	4.443×10^6	1.474×10^6	1.310×10^7	0	NA	N
GPDI×B.	Gross Private Direct Investment (Q)	2770	1696	4671	0	NA	N
Imports	Imp. of Goods & Services as % of GDP (Q)	15.02	11.90	18.20	0	NA	N
Exports	Exp. of Goods & Services as % of GDP (Q)	11.39	8.90	13.80	0	NA	N
MACROECONOMIC SOURCED VARIABLES (SOURCE: BLOOMBERG)							
PMI Composite	Purchasing Managers Index C. (M)	54.67	37.90	66.90	0	NA	Y
PMI Manufacturing	Purchasing Managers Index M. (M)	52.94	34.50	63.80	0	NA	Y
MACROECONOMIC SOURCED VARIABLES (SOURCE: OECD)							
GDP Growth FC	Year ahead GDP Growth Forecast	2.334	-29.857	35.317	0	NA	Y
OTHER MACROECONOMIC VARIABLES (SOURCE: NO SOURCE)							

Election Year	Indicator function El. Y. (1) vs. No El. Y. (0)	0.238	0	1	0	NA	Y
MACROECONOMIC TRANSFORMED VARIABLES							
Private Debt To GDP	Private Debt / GDP (Q)	80.83	64.58	98.20	0	NA	Y
Trade Openness	Imports + Exports (Q)	26.41	21.20	31.20	0	NA	Y
Real GDP P.C. Growth	R. GDP P.C. Growth vs. Prev. Q. (Q)	0.347	-8.91	7.510	0	NA	Y
Oil Price Change	Oil Price Change Comp. to Prev. D. (D)	0.024	-64.370	41.202	0	NA	Y
M1 Yearly Change	M1 Supply Change Comp. to Prev. Y. (W)	16.256	-6.528	162.091	0	NA	Y
FDI Change	FDI Change Comp. to Prev. Q. (Q)	2.025	-30.448	26.787	0	NA	Y
GPDI Change	GPDI Change Comp. to Prev. Q. (Q)	1.017	-16.741	16.894	0	NA	Y

All variables (except for the Log Loss Return and the Election Year variables) were lagged by one day before using them in the estimation of the VaR.

B Alternate Data Generating Processes

B.1 Gnecco Simulation Alternate DGPs

The alternate DGPs of the Gnecco simulation study, which were used to test the consistency of the methods (see Section 3.5), were generated by changing the shape parameter $\xi(x) = \frac{1}{v}$ through the number of degrees of freedom v. The original DGP used $\xi(x) = \frac{1}{4}$. The three alternate variations are given by:

$$\begin{aligned}
\xi_1(x) &= \frac{1}{6}, \\
\xi_2(x) &= \frac{1}{2}, \\
\xi_3(x) &= \frac{5}{6}.
\end{aligned}$$

B.2 Ahmed Simulation Alternate DGPs

The alternate DGPs of the Ahmed simulation study, which were used to test the consistency of the methods (see Section 3.5), were generated by changing the shape parameter function $\xi(x)$. The original DGP used $\xi(x) = 0.15 + 0.7 \times \mathbb{1}\{x_3 = 2\} + 0.93 \times \mathbb{1}\{x_2 = 2\}$. The three alternative variations are given by:

$$\begin{aligned}
\xi_1(x) &= 0.00 + 0.15 \times \mathbb{1}\{x_3 = 2\} + 0.15 \times \mathbb{1}\{x_2 = 2\}, \\
\xi_2(x) &= 0.15 + 0.15 \times \mathbb{1}\{x_3 = 2\} + 0.30 \times \mathbb{1}\{x_2 = 2\}, \\
\xi_3(x) &= 0.30 + 0.30 \times \mathbb{1}\{x_3 = 2\} + 0.30 \times \mathbb{1}\{x_2 = 2\}.
\end{aligned}$$

B.3 Time-Invariant Financial Market Simulation Alternate DGPs

The alternate DGPs of the Time-Invariant Financial Market simulation study, which were used to test the consistency of the methods (see Section 3.5), were generated by changing the shape parameter function $\xi(x) = \frac{1}{v(x)}$. The original DGP used $v(x) = 3 \times \left[2 + \tanh\left(-2\left(\frac{4}{7}x_1 + \frac{2}{7}x_2 + \frac{1}{7}x_3\right)\right)\right]$. The three alternative variations are given by:

$$\begin{aligned}
v_1(x) &= 4 \times \left[2 + \tanh\left(-2\left(\frac{4}{7}x_1 + \frac{2}{7}x_2 + \frac{1}{7}x_3\right)\right)\right], \\
v_2(x) &= 2 \times \left[2 + \tanh\left(-2\left(\frac{4}{7}x_1 + \frac{2}{7}x_2 + \frac{1}{7}x_3\right)\right)\right], \\
v_3(x) &= 1 \times \left[2 + \tanh\left(-2\left(\frac{4}{7}x_1 + \frac{2}{7}x_2 + \frac{1}{7}x_3\right)\right)\right].
\end{aligned}$$

C Hyperparameter Settings

For each method, we only tuned certain hyperparameters (to be discussed in following sections). For the parameters we tuned we will present the range of configurations that were tried and the final hyperparameter settings obtained through cross-validation. The other hyperparameters were set to either the default value given in the R package or a default value determined by some initial model runs. The latter will be presented in the following sections as well.

C.1 ERF Hyperparameter Settings

For the ERF method, we tuned the minimum node size (κ)), the lambda parameter (λ)), and the intermediate quantile (τ_0). The first two were optimized for both the simulation studies and the data application, whereas the intermediate quantile was only optimized for the simulation studies. For the data application, we could not properly use the cross-validation scheme described in Section 2.3.5, therefore we set $\tau_0 = 0.8$ which is the default value in the ERF package.

This means that for the simulation studies we tried the following range of parameters:

- $\kappa \in \{5, 10, 40, 80, 100\}$,
- $\lambda \in \{0, 0.001, 0.01, 0.1\}$,
- $\tau_0 \in \{0.8, 0.85, 0.90\}$.

For the data application we tried the following range of parameters:

- $\kappa \in \{5, 10, 40, 80, 100\}$,
- $\lambda \in \{0, 0.001, 0.01, 0.1\}$,
- $\tau_0 = 0.8$.

Gnecco Simulation ERF Hyperparameter Settings:

- $\kappa = 100$,
- $\lambda = 0.001$,
- $\tau_0 = 0.8$.

Ahmed Simulation ERF Hyperparameter Settings:

- $\kappa = 100$,
- $\lambda = 0.001$,
- $\tau_0 = 0.9$.

Time-Invariant Financial Market Simulation ERF Hyperparameter Settings:

- $\kappa = 40$,
- $\lambda = 0$,
- $\tau_0 = 0.85$.

Time-Varying Financial Market Simulation ERF Hyperparameter Settings (non-filtered):

- $\kappa = 40$,
- $\lambda = 0$,
- $\tau_0 = 0.85$.

Time-Varying Financial Market Simulation ERF Hyperparameter Settings (filtered):

- $\kappa = 40$,
- $\lambda = 0.001$,
- $\tau_0 = 0.9$.

S&P500 Application ERF Hyperparameter Settings (non-filtered):

- $\kappa = 80$,
- $\lambda = 0.1$,
- $\tau_0 = 0.8$.

S&P500 Application ERF Hyperparameter Settings (filtered):

- $\kappa = 40$,
- $\lambda = 0.1$,
- $\tau_0 = 0.8$.

C.2 ERF Pareto Hyperparameter Settings

For the ERF Pareto method, we tuned the minimum node size (κ)), the lambda parameter (λ)), and the intermediate quantile (τ_0). The first two were optimized for both the simulation studies and the data application, whereas the intermediate quantile was only optimized for the simulation studies. For the data application, we could not properly use the cross-validation scheme described in Section 2.3.5, therefore we chose to set $\tau_0 = 0.9$ since this was the most observed τ_0 setting in the simulation studies. We set the number of trees to 2000 after some initial running, this number of trees is in alignment with the advised setting for the ERF method as advised by Gnecco et al. (2022).

This means that for the simulation studies we tried the following range of parameters:

- $\kappa \in \{5, 10, 40, 80, 100\}$,
- $\lambda \in \{0, 0.001, 0.01, 0.1\}$,
- $\tau_0 \in \{0.8, 0.85, 0.90\}$.

For the data application we tried the following range of parameters:

- $\kappa \in \{5, 10, 40, 80, 100\}$,
- $\lambda \in \{0, 0.001, 0.01, 0.1\}$,
- $\tau_0 = 0.9$.

Gnecco Simulation ERF Pareto Hyperparameter Settings:

- $\kappa = 40$,
- $\lambda = 0.1$,
- $\tau_0 = 0.9$.

Ahmed Simulation ERF Pareto Hyperparameter Settings:

- $\kappa = 80$,
- $\lambda = 0.1$,
- $\tau_0 = 0.9$.

Time-Invariant Financial Market Simulation ERF Pareto Hyperparameter Settings:

- $\kappa = 5$,
- $\lambda = 0.1$,
- $\tau_0 = 0.9$.

Time-Varying Financial Market Simulation ERF Pareto Hyperparameter Settings (non-filtered):

- $\kappa = 40$,
- $\lambda = 0.1$,
- $\tau_0 = 0.9$.

Time-Varying Financial Market Simulation ERF Pareto Hyperparameter Settings (filtered):

- $\kappa = 5$,
- $\lambda = 0.1$,
- $\tau_0 = 0.9$.

S&P500 Application ERF Pareto Hyperparameter Settings (non-filtered):

- $\kappa = 40$,
- $\lambda = 0.1$,
- $\tau_0 = 0.9$.

S&P500 Application ERF Pareto Hyperparameter Settings (filtered):

- $\kappa = 10$,
- $\lambda = 0.01$,
- $\tau_0 = 0.9$.

C.3 Ahmed Hyperparameter Settings

For the Ahmed method, we tuned the minimum node size for the regression tree (κ_r), the minimum node size for the classification tree (κ_c)), and the fixed value threshold (u). For the first two we cross-validated over a fixed range of options, whereas for the fixed threshold u, we followed Ahmed (2022) by considering the starting points of the tail region at the 0.7, 0.8, and 0.9 historical quantiles. In turn, we tuned the fixed value threshold using the cross-validation scheme described in 2.4.5 over the three possible thresholds for both the simulation studies and the data application.

Additionally, we set the number of trees to 500 for the regression forest and 1000 for the classification forest after some initial running.

This means that we tried the following range of parameters:

- $\kappa_r \in \{10, 20, 50, 100, 250, 500, 1000\}$,
- $\kappa_c \in \{10, 20, 50, 100, 250, 500, 1000\}$,
- $u \in \{u_1 = Y_{train}(\tau = 0.7), u_2 = Y_{train}(\tau = 0.8), u_3 Y_{train}(\tau = 0.9)\}$.

Gnecco Simulation Ahmed Hyperparameter Settings:

- $\kappa_r = 250$,
- $\kappa_c = 10$,
- $u = 3.5$.

Ahmed Simulation Ahmed Hyperparameter Settings:

- $\kappa_r = 250$,
- $\kappa_c = 100$,
- $u = 20$.

Time-Invariant Financial Market Simulation Ahmed Hyperparameter Settings:

- $\kappa_r = 500$,
- $\kappa_c = 20$,
- $u = 2$.

Time-Varying Financial Market Simulation Ahmed Hyperparameter Settings (non-filtered):

- $\kappa_r = 1000$,
- $\kappa_c = 1000$,
- $u = 0.1$.

Time-Varying Financial Market Simulation Ahmed Hyperparameter Settings (filtered):

- $\kappa_r = 500$,
- $\kappa_c = 20$,
- $u = 2$.

S&P500 Application Ahmed Hyperparameter Settings (non-filtered):

- $\kappa_r = 250$,
- $\kappa_c = 20$,
- $u = 2$.

S&P500 Application Ahmed Hyperparameter Settings (filtered):

- $\kappa_r = 20$,
- $\kappa_c = 1000$,
- $u = 1.5$.