The importance of higher moments in VaR and cVaR estimation.

Faes E.[1] Mertens de Wilmars S.[2] Pratesi F.[3]

Antwerp Management School

Prof. dr. Annaert Prof. dr. De Ceuster Prof. dr. Zhang

Master in Finance

June 2021

[1]Enjo.Faes@student.ams.ac.be
[2]Stephane.MertensdeWilmars@student.ams.ac.be
[3]Filippo.Pratesi@student.ams.ac.be

For our families and loved ones

Acknowledgements

First of all, many thanks to our families and loved ones that supported us during the writing of this thesis. Secondly, thank you professors Zhang, Annaert and De Ceuster for the valuable insights you have given us in preparation of this thesis and the many questions answered. We must be grateful for the classes of R programming by prof Zhang.

Secondly, we have to thank the developer of the software we used for our thesis. A profuse thanks to Allaire, the founder and CEO of RStudio. Thanks for making data science easier, more accessible and fun. We must also be grateful to Gruber for inventing “Markdown”, to MacFarlane for creating “Pandoc” which converts Markdown to a large number of output formats, and to Xie for creating “knitr” which introduced R Markdown as a way of embedding code in Markdown documents, and “bookdown” which added tools for technical and longer-form writing.Special thanks to Ismay, who created the “thesisdown” package that helped many PhD students write their theses in R Markdown. And a very special thanks to McManigle, whose adaption of Evans’ adaptation of Gillow’s original maths template for writing an Oxford University DPhil thesis in “LaTeX” provided the template that Ulrik Lyngs in turn adapted for R Markdown, which we also owe a big thank you. Without which this thesis could not have been written in this format (Lyngs 2019).

Finally, we thank Ghalanos (2020b) for making the implementation of GARCH models integrated in R via his package “Rugarch”. By doing this, he facilitated the process of understanding the whole process and doing the analysis for our thesis.

Enjo Faes,
Stephane Mertens de Wilmars,
Filippo Pratesi
Antwerp Management School, Antwerp
27 June 2021

Abstract

The greatest abstract all times

Contents

List of Figures

List of Tables

List of Abbreviations

ACD . . . . . . Autoregressive Conditional Density models (Hansen, 1994)

ARCH . . . . . Autoregressive Conditional Heteroscedasticity model (Bollerslev, 1986)

GARCH . . . Generalized Autoregressive Conditional Heteroscedasticity model (Bollerslev, 1986)

IGARCH . . . Integrated GARCH (Bollerslev, 1986)

EGARCH . . Exponential GARCH (Nelson, 1991)

GJRGARCH Glosten-Jagannathan-Runkle GARCH model (Glosten et al. 1993)

NAGARCH . Nonlinear asymmetric GARCH (Engle and Ng, 1993)

TGARCH . . Threshold GARCH (Zakoian, 1994)

TSGARCH . . Also called Absolute Value GARCH or AVGARCH referring to Taylor (1986) and Schwert (1989)

EWMA . . . . Exponentially Weighted Moving Average model

i.i.d, iid . . . . Independent and identically distributed

T . . . . Student's T-distribution

ST . . . . Skewed Student's T-distribution

SGT . . . . . . Skewed Generalized T-distribution

GED . . . . . . Generalized Error Distribution

SGED . . . . . Skewed Generalized Error Distribution

NORM . . . . Normal distribution

VaR . . . . . . Value-at-Risk

cVaR . . . . . Expected shortfall or conditional Value-at-Risk

Introduction

A general assumption in finance is that stock returns are normally distributed. However, various authors have shown that this assumption does not hold in practice: stock returns are not normally distributed (Officer 1972). For example, Theodossiou (2000) mentions that “empirical distributions of log-returns of several financial assets exhibit strong higher-order moment dependencies which exist mainly in daily and weekly log-returns and prevent monthly, bimonthly and quarterly log-returns from obeying the normality law implied by the central limit theorem. As a consequence, price changes do not follow the geometric Brownian motion.” So in reality, stock returns exhibit fat-tails and peakedness (Officer 1972), these are some of the so-called stylized facts of returns.

Additionally, a point of interest is the predictability of stock prices. Fama (1965) explains that the question in academic and business circles is: “To what extent can the past history of a common stock’s price be used to make meaningful predictions concerning the future price of the stock?”. There are two viewpoints towards the predictability of stock prices. Firstly, some argue that stock prices are unpredictable or very difficult to predict by their past returns (i.e. have very little serial correlation) because they simply follow a Random Walk process (Fama 1970). On the other hand, Lo & MacKinlay mention that “financial markets *are* predictable to some extent but far from being a symptom of inefficiency or irrationality, predictability is the oil that lubricates the gears of capitalism”. Furthermore, there is also no real robust evidence for the predictability of returns themselves, let alone be out-of-sample (Welch and Goyal 2008). This makes it difficult for corporations to manage market risk, i.e. the variability of stock prices.

Risk, in general, can be defined as the volatility of unexpected outcomes (Jorion 2007). The measure Value at Risk (VaR), developed in response to the financial disaster events of the early 1990s, has been very important in the financial world. Corporations have to manage their risks and thereby include a future risk measurement. The tool of VaR has now become a standard measure of risk for many financial institutions going from banks, that use VaR to calculate the adequacy of their capital structure, to other financial services companies to assess the exposure of their positions and portfolios. The 5% VaR can be informally defined as the maximum loss of a portfolio, during a time horizon, excluding all the negative events with a combined probability lower than 5% while the Conditional Value at Risk (CVaR) can be informally defined as the average of the events that are lower than the VaR. Bali, Mo, et al. (2008) explains that many implementations of the CVaR have the assumption that asset and portfolio's returns are normally distributed but that it is an inconsistency with the evidence empirically available which outlines a more skewed distribution with fatter tails than the normal. This lead to the conclusion that the assumption of normality, which simplifies the computation of VaR, can bring to incorrect numbers, underestimating the probability of extreme events happening.

This paper has the aim to replicate and update the research made by Bali, Mo, et al. (2008) on US indexes, analyzing the dynamics proposed with a European outlook. The main contribution of the research is to provide the industry with a new approach to calculating VaR with a flexible tool for modeling the empirical distribution of returns with higher accuracy and characterization of the tails.

The paper is organized as follows. Chapter 1 discusses at first the alternative distribution than the normal that we are going to evaluate during the analysis (Student's t-distribution, Generalized Error Distribution, Skewed t-distribution, Skewed Generalized Error Distribution, Skewed Generalized t-distribution), then the discrete time GARCH models used (GARCH, IGARCH, EGARCH, GJRGARCH, NAGARCH, TGARCH, TSGARCH or AVGARCH and EWMA) are presented as extensions of the Engle (1982) 's ARCH model. Chapter 2 describes the dataset used and the methodology followed in modeling the volatility with the GARCH model by Bollerslev (1986) and with its refinements using Maximum likelihood estimation to find the distribution parameters. Then a description is given of how are performed the control tests (un- and conditional coverage test, dynamic quantile test) used in the paper to evaluate the performances of the different GARCH models and underlying distributions. In chapter 3, findings are presented and discussed, in chapter 4 the findings of the performed tests are shown and interpreted and in chapter 5 the investigation and the results are summarized.

1

Literature review

1.1 Stylized facts of returns

When analyzing returns as a time-series, we look at log returns. The log returns are similar to simple returns so the stylized facts of returns apply to both. One assumption that is made often in financial applications is that returns are iid, or independently and identically distributed, another is that they are normally distribution. Are these valid assumptions? Below the stylized facts[1] following Annaert (2021) for returns are given.

- Returns are *small and volatile* (with the standard deviation being larger than the mean on average)
- Returns have very little serial correlation as mentioned by for example Bollerslev (1987).
- Returns exhibit conditional heteroskedasticity, or *volatility clustering*. There is no constant variance, but it is time-varying (homoskedasticity). Bollerslev (1987) describes it as "rates of return data are characterized by volatile and tranquil periods".

[1]Stylized facts are the statistical properties that appear to be present in many empirical asset returns (across time and markets)

- Returns also exhibit *asymmetric volatility*, in that sense volatility increases more after a negative return shock than after a large positive return shock. This is also called the *leverage effect.*

- Returns are *not normally distributed* which is also one of the conclusions by Fama (1965). Returns have tails fatter than a normal distribution (leptokurtosis) and thus are riskier than under the normal distribution. Log returns **can** be assumed to be normally distributed. However, this will be examined in our empirical analysis if this is appropriate. This makes that simple returns follow a log-normal distribution, which is a skewed density distribution.

Firms holding a portfolio have a lot of things to consider: expected return of a portfolio, the probability to get a return lower than some threshold, the probability that an asset in the portfolio drops in value when the market crashes. All the previous requires information about the return distribution or the density function. What we know from the stylized facts of returns that the normal distribution is not appropriate for returns. Below we summarize some alternative distributions that could be a better approximation of returns than the normal one.

1.1.1 Alternative distributions than the normal

Student's t-distribution

A common alternative for the normal distribution is the Student t distribution. Similarly to the normal distribution, it is also symmetric (skewness is equal to zero). The probability density function (pdf), again following Annaert (2021), is given by equation (1.1). As will be seen in 1.2, GARCH models are used for volatility modeling in practice. Bollerslev (1987) examined the use of the GARCH-Student or GARCH-t model as an alternative to the standard Normal distribution, which relaxes the assumption of conditional normality by assuming the standardized innovation to follow a standardized Student t-distribution (Bollerslev 2008).

$$f(x) = \frac{\Gamma(\frac{n+1}{2})}{\Gamma(\frac{n}{2})\sqrt{\pi n}}(1 + \frac{x^2}{n})^{-(n+1)/2} \tag{1.1}$$

As can be seen the pdf depends on the degrees of freedom n. To be consistent with Ghalanos (2020a), the following general equation is used for the pdf (1.2).

$$f(x) = \frac{\Gamma(\frac{\nu+1}{2})}{\Gamma(\frac{\nu}{2})\sqrt{\sigma\pi\nu}}\left(1 + \frac{(x-\mu)^2}{\sigma\nu}\right)^{-(\nu+1)/2} \tag{1.2}$$

where μ, σ and ν are respectively the location, scale and shape (tail-thickness) parameters. The symbol Γ is the Gamma function.

Unlike the normal distribution, which depends entirely on two moments only, the student t distribution has fatter tails (thus it has a kurtosis coefficient), if the degrees of freedom are finite. This kurtosis coefficient is given by equation (1.3). This is useful while as already mentioned, the standardized residuals appear to have fatter tails than the normal distribution following Bollerslev (2008).

$$kurt = 3 + \frac{6}{\nu - 4} \tag{1.3}$$

Generalized Error Distribution

The GED distribution is nested in the generalized t distribution by McDonald and Newey (1988) is used in the GED-GARCH model by Nelson (1991) to model stock market returns. This model replaced the assumption of conditional normally distributed error terms by standardized innovations that following a generalized error distribution. It is a symmetric, unimodal distribution (location parameter is the mode, median and mean). This is also sometimes called the exponential power distribution (Bollerslev 2008). The conditional density (pdf) is given by equation (1.4) following Ghalanos (2020a).

$$f(x) = \frac{pe^{\left|\frac{x-\mu}{\sigma}\right|^{p}}}{2^{1+p^{(-1)}}\sigma\Gamma(p^{-1})} \tag{1.4}$$

where μ, σ and p are respectively the location, scale and shape parameters .

Skewed t-distribution

The density function can be derived following Fernández and Steel (1998) who showed how to introduce skewness into uni-modal standardized distributions (Trottier and Ardia 2015). The first equation from Trottier and Ardia (2015), here equation (1.5) presents the skewed t-distribution.

$$f_\xi(z) \equiv \frac{2\sigma_\xi}{\xi+\xi^{-1}} f_1\left(z_\xi\right), \quad z_\xi \equiv \begin{cases} \xi^{-1}\left(\sigma_\xi z + \mu_\xi\right) & \text{if } z \geq -\mu_\xi/\sigma_\xi \\ \xi\left(\sigma_\xi z + \mu_\xi\right) & \text{if } z < -\mu_\xi/\sigma_\xi \end{cases} \tag{1.5}$$

where $\mu_\xi \equiv M_1\left(\xi - \xi^{-1}\right), \quad \sigma_\xi^2 \equiv \left(1 - M_1^2\right)\left(\xi^2 + \xi^{-2}\right) + 2M_1^2 - 1, \quad M_1 \equiv 2\int_0^\infty u f_1(u)du$ and ξ between 0 and ∞. $f_1(\cdot)$ is in this case equation (1.1), the pdf of the student t distribution.

According to Giot and Laurent (2003); Giot and Laurent (2004), the skewed t-distribution outperforms the symmetric density distributions.

Skewed Generalized Error Distribution

What also will be interesting to examine is the SGED distribution of Theodossiou (2000) in GARCH models, as in the work of Lee et al. (2008). The SGED distribution extends the Generalized Error Distribution (GED) to allow for skewness and leptokurtosis. The density function can be derived following Fernández and Steel (1998) who showed how to introduce skewness into uni-modal standardized distributions (Trottier and Ardia 2015). It can also be found in Theodossiou (2000). The pdf is then given by the same equation (1.5) as the skewed t-distribution but with $f_1(\cdot)$ equal to equation (1.4).

Skewed Generalized t-distribution

The SGT distribution of introduced by Theodossiou (1998) and applied by Bali and Theodossiou (2007) and Bali, Mo, et al. (2008). According to Bali, Mo, et al. (2008) the proposed solutions (use of historical simulation, student's t-distribution, generalized error distribution or a mixture of two normal distributions) to the non-normality of standardized financial returns only partially solved the issues of skewness and leptokurtosis. The density of the generalized t-distribution of McDonald and Newey (1988) is given by equation (1.6) (Bollerslev et al. 1994).

$$f\left[\varepsilon_t \sigma_t^{-1}; p, \psi\right] = \frac{p}{2\sigma_t \;\cdot \psi^{1/p} B(1/p, \psi) \cdot \left[1 + |\varepsilon_t|^p \,/\, (\psi b^p \sigma_t^p)\right]^{\psi + 1/p}} \tag{1.6}$$

where $B(1/\eta, \psi)$ is the beta function $(=\Gamma(1/\eta)\Gamma(\psi)\Gamma(1/\eta+\psi))$, $\psi\eta > 2,\ \eta > 0 \ \textit{and}\ \psi > 0$, $\beta = [\Gamma(\psi)\Gamma(1/\eta)/\Gamma(3/\eta)\Gamma(\psi-2/\eta)]^{1/2})$, the scale factor and one shape parameter p.

Again the skewed variant is given by equation (1.5) but with $f_1(\cdot)$ equal to equation (1.6) following Trottier and Ardia (2015).

1.2 Volatility modeling

1.2.1 Rolling volatility

When volatility needs to be estimated on a specific trading day, the method used as a descriptive tool would be to use rolling standard deviations. Engle (2001) explains the calculation of rolling standard deviations, as the standard deviation over a fixed number of the most recent observations. For example, for the past month it would then be calculated as the equally weighted average of the squared deviations from the mean (i.e. residuals) from the last 22 observations (the average amount of trading or business days in a month). All these deviations are thus given an equal weight. Also, only a fixed number of past recent observations is examined. Engle regards this formulation as the first ARCH model.

1.2.2 ARCH model

Autoregressive Conditional Heteroscedasticity (ARCH) models, proposed by Engle (1982), was in the first case not used in financial markets but on inflation. Since then, it has been used as one of the workhorses of volatility modeling. To fully capture the logic behind GARCH models, the building blocks are examined in the first place. There are three building blocks of the ARCH model: returns, the innovation process and the variance process (or volatility function), written out in respectively equation (1.7), (1.8) and (1.9). Returns are written as a constant part (μ) and an unexpected part, called noise or the innovation process. The innovation process is the volatility (σ_t) times z_t, which is an independent identically distributed random variable with a mean of 0 (zero-mean) and a variance of 1 (unit-variance). The independent from iid, notes the fact that the z-values are not correlated, but completely independent of each other. The distribution is not yet assumed. The third component is the variance process or the expression for the volatility. The variance is given by a constant ω, plus the random part which depends on the return shock of the previous period squared (ε_{t-1}^2). In that sense when the uncertainty or surprise in the last period increases, then the variance becomes larger in the next period. The element σ_t^2 is thus known at time $t-1$, while it is a deterministic function of a random variable observed at time $t-1$ (i.e. ε_{t-1}^2).

$$y_t = \mu + \varepsilon_t \tag{1.7}$$

$$\varepsilon_t = \sigma_t * z_t,\ where\ z_t \overset{iid}{\sim} (0,1) \tag{1.8}$$

$$\sigma_t^2 = \omega + \alpha_1 * \varepsilon_{t-1}^2 \tag{1.9}$$

From these components we could look at the conditional moments (or expected returns and variance). We can plug in the component σ_t into the conditional mean innovation ε_t and use the conditional mean innovation to examine the conditional mean return. In equation (1.10) and (1.11) they are derived. Because the random variable z_t is distributed with a zero-mean, the conditional expectation is 0. As a consequence, the conditional mean return in equation (1.11) is equal to the unconditional mean in the most simple case. But variations are possible using ARMA (eg. AR(1)) processes.

$$\mathbb{E}_{t-1}(\varepsilon_t) = \mathbb{E}_{t-1}(\sqrt{\omega + \alpha_1 * \varepsilon_{t-1}^2} * z_t) = \sigma_t \mathbb{E}_{t-1}(z_t) = 0 \tag{1.10}$$

$$\mathbb{E}_{t-1}(y_t) = \mu + \mathbb{E}_{t-1}(\varepsilon_t) = \mu \tag{1.11}$$

For the conditional variance, knowing everything that happened until and including period $t-1$ the conditional innovation variance is given by equation (1.12). This is equal to σ_t^2, while the variance of z_t is equal to 1. Then it is easy to derive the conditional variance of returns in equation (1.13), that is why equation (1.9) is called the variance equation.

$$var_{t-1}(\varepsilon_t) = \mathbb{E}_{t-1}(\varepsilon_t^2) = \mathbb{E}_{t-1}(\sigma_t^2 * z_t^2) = \sigma_t^2 \mathbb{E}_{t-1}(z_t^2) = \sigma_t^2 \tag{1.12}$$

$$var_{t-1}(y_t) = var_{t-1}(\varepsilon_t) = \sigma_t^2 \tag{1.13}$$

The unconditional variance is also interesting to derive, while this is the long-run variance, which will be derived in (1.17). After deriving this using the law of iterated expectations and assuming stationarity for the variance process, one would get (1.14) for the unconditional variance, equal to the constant c and divided by $1 - \alpha_1$, the slope of the variance equation.

$$\sigma^2 = \frac{\omega}{1-\alpha_1} \tag{1.14}$$

This leads to the properties of ARCH models. - Stationarity condition for variance: $\omega > 0$ and $0 \leq \alpha_1 < 1$.

- Zero-mean innovations
- Uncorrelated innovations

Thus a weak white noise process ε_t.

Stationarity implies that the series on which the ARCH model is used does not have any trend and has a constant expected mean. Only the conditional variance is changing.

The unconditional 4th moment, kurtosis $\mathbb{E}(\varepsilon_t^4)/\sigma^4$ of an ARCH model is given by equation (1.15). This term is larger than 3, which implicates that the fat-tails (a stylized fact of returns).

$$3\frac{1-\alpha_1^2}{1-3\alpha_1^2} \tag{1.15}$$

Another property of ARCH models is that it takes into account volatility clustering. Because we know that $var(\varepsilon_t) = \mathbb{E}(\varepsilon_t^2) = \sigma^2 = \omega/(1-\alpha_1)$, we can plug in ω for the conditional variance $var_t(\varepsilon_{t+1}) = \mathbb{E}(\varepsilon_{t+1}^2) = \sigma_{t+1}^2 = c + \alpha_1 * \varepsilon_t^2$. Thus it follows that equation (1.16) displays volatility clustering. If we examine the RHS, as $\alpha_1 > 0$ (condition for stationarity), when shock ε_t^2 is larger than what you expect it to be on average σ^2 the LHS will also be positive. Then the conditional variance will be larger than the unconditional variance. Briefly, large shocks will be followed by more large shocks.

$$\sigma_{t+1}^2 - \sigma^2 = \alpha_1 * (\varepsilon_t^2 - \sigma^2) \tag{1.16}$$

Excess kurtosis can be modeled, even when the conditional distribution is assumed to be normally distributed. The third moment, skewness, can be introduced using a skewed conditional distribution as we saw in part 1.1.1. The serial correlation for squared innovations is positive if fourth moment exists (equation (1.15), this is volatility clustering once again.

The estimation of ARCH model and in a next step GARCH models will be explained in the methodology. However how will then the variance be forecasted? Well, the conditional variance for the k-periods ahead , denoted as period $T+k$, is given by equation (1.17). This can already be simplified, while we know that $\sigma^2_{T+1} = \omega + \alpha_1 * \varepsilon^2_T$ from equation (1.9).

$$\begin{aligned} \mathbb{E}_T(\varepsilon^2_{T+k}) &= \omega * (1 + \alpha_1 + ... + \alpha^{k-2}) + \alpha^{k-1} * \sigma^2_{T+1} \\ &= \omega * (1 + \alpha_1 + ... + \alpha^{k-1}) + \alpha^k * \sigma^2_T \end{aligned} \tag{1.17}$$

It can be shown that then the conditional variance in period $T+k$ is equal to equation (1.18). The LHS is the predicted conditional variance k-periods ahead above its unconditional variance, σ^2. The RHS is the difference current last-observed return residual ε^2_T above the unconditional average multiplied by α_1^k, a decreasing function of k (given that $0 \leq \alpha_1 < 1$). The further ahead predicting the variance, the closer α_1^k comes to zero, the closer to the unconditional variance, i.e. the long-run variance.

$$\mathbb{E}_T(\varepsilon^2_{T+k}) - \sigma^2 = \alpha_1^k * (\varepsilon^2_T - \sigma^2) \tag{1.18}$$

1.2.3 Univariate GARCH models

An improvement on the ARCH model is the Generalised Autoregressive Conditional Heteroscedasticity (GARCH). This model and its variants come in to play because of the fact that calculating standard deviations through rolling periods, gives an equal weight to distant and nearby periods, by such not taking into account empirical evidence of volatility clustering, which can be identified as positive autocorrelation in the absolute returns. GARCH models are an extension to ARCH models, as they incorporate both a novel moving average term (not included in ARCH) and the autoregressive component.

All the GARCH models below are estimated using the package rugarch by Ghalanos (2020b). We use specifications similar to Ghalanos (2020a). Parameters have to be restricted so that the variance output always is positive, except for the EGARCH model, as this model does not mathematically allow for a negative output. An overview (of a selection) of GARCH models is given in the following table.

Table 1.1: GARCH models, the founders

Author(s)/user(s)	Model
Engle (1982)	ARCH model
Bollerslev (1986)	GARCH model
Bollerslev (1986)	IGARCH model
Nelson (1991)	EGARCH model
Glosten et al. (1993)	GJRGARCH model
Engle and Ng (1993)	NAGARCH model
Zakoian (1994)	TGARCH model
Taylor (1986) and Schwert (1989)	TSGARCH (or AVGARCH) model
Morgan Guaranty Trust Company (1996)	EWMA or RiskMetrics model

GARCH model

The standard GARCH model (Bollerslev 1986) is written consistent with Ghalanos (2020a) as in equation (1.19) without external regressors.

$$\sigma_t^2 = \omega + \sum_{j=1}^{q} \alpha_j \varepsilon_{t-j}^2 + \sum_{j=1}^{p} \beta_j \sigma_{t-j}^2 \tag{1.19}$$

where σ_t^2 denotes the conditional variance, ω the intercept and ε_t^2 the residuals from the used mean process. The GARCH order is defined by (q, p) (ARCH, GARCH). As Ghalanos (2020a) describes: "one of the key features of the observed behavior of financial data which GARCH models capture is volatility clustering which may be quantified in the persistence parameter $\hat{P}$" specified as in equation (1.20).

$$\hat{P} = \sum_{j=1}^{q} \alpha_j + \sum_{j=1}^{p} \beta_j. \tag{1.20}$$

The unconditional variance of the standard GARCH model of Bollerslev is very similar to the ARCH model, but with the Garch parameters (β's) included as in equation (1.21).

$$\begin{aligned} \hat{\sigma}^2 &= \frac{\hat{\omega}}{1 - \hat{P}} \\ &= \frac{\hat{\omega}}{1 - \alpha - \beta} \end{aligned} \tag{1.21}$$

IGARCH model

Following Ghalanos (2020a), the integrated GARCH model (Bollerslev 1986) can also be estimated. This model assumes the persistence $\hat{P} = 1$. This is done by Ghalanos, by setting the sum of the ARCH and GARCH parameters to 1. Because of this unit-persistence, the unconditional variance cannot be calculated.

EGARCH model

The EGARCH model or exponential GARCH model (Nelson 1991) is defined as in equation (1.22). The advantage of the EGARCH model is that there are no parameter restrictions, since the output is log variance (which cannot be negative mathematically), instead of variance.

$$\log_e(\sigma_t^2) = \omega + \sum_{j=1}^{q}(\alpha_j z_{t-j} + \gamma_j(|z_{t-j}| - E|z_{t-j}|)) + \sum_{j=1}^{p}\beta_j \log_e(\sigma_{t-j}^2) \tag{1.22}$$

where α_j captures the sign effect and γ_j the size effect.

GJRGARCH model

The GJRGARCH model (Glosten et al. 1993) models both positive as negative shocks on the conditional variance asymmetrically by using an indicator variable I, it is specified as in equation (1.23).

$$\sigma_t^2 = \omega + \sum_{j=1}^{q}(\alpha_j \varepsilon_{t-j}^2 + \gamma_j I_{t-j}\varepsilon_{t-j}^2) + \sum_{j=1}^{p}\beta_j \sigma_{t-j}^2 \tag{1.23}$$

where γ_j represents the *leverage* term. The indicator function I takes on value 1 for $\varepsilon \leq 0$, 0 otherwise. Because of the indicator function, persistence of the model now crucially depends on the asymmetry of the conditional distribution used according to Ghalanos (2020a).

NAGARCH model

The NAGARCH or nonlinear asymmetric model (Engle and Ng 1993). It is specified as in equation (1.24). The model is *asymmetric* as it allows for positive and negative shocks to differently affect conditional variance and *nonlinear* because a large shock is not a linear transformation of a small shock.

$$\sigma_t^2 = \omega + \sum_{j=1}^{q}\alpha_j(\varepsilon_{t-j} + \gamma_j\sqrt{\sigma_{t-j}})^2 + \sum_{j=1}^{p}\beta_j \sigma_{t-j}^2 \tag{1.24}$$

As before, γ_j represents the *leverage* term.

TGARCH model

The TGarch or threshold model (Zakoian 1994) also models assymetries in volatility depending on the sign of the shock, but contrary to the GJRGARCH model it uses the conditional standard deviation instead of conditional variance. It is specified as in (1.25).

$$\sigma_t = \omega + \sum_{j=1}^{q}(\alpha_j^+ \varepsilon_{t-j}^+ \alpha_j^- + \varepsilon_{t-j}^-) + \sum_{j=1}^{p} \beta_j \sigma_{t-j} \tag{1.25}$$

where ε_{t-j}^+ is equal to ε_{t-j} if the term is negative and equal to 0 if the term is positive. The reverse applies to ε_{t-j}^-. They cite Davidian and Carroll (1987) who find that using volatility instead of variance as scaling input variable gives better variance estimates. This is due to absolute residuals (contrary to squared residuals with variance) more closely predicting variance for non-normal distributions.

TSGARCH model

The absolute value Garch model or TS-Garch model, as named after Taylor (1986) and Schwert (1989), models the conditional standard deviation and is intuitively specified like a normal GARCH model, but with the absolute value of the shock term. It is specified as in (1.26).

$$\sigma_t = \omega + \sum_{j=1}^{q}(\alpha_j \left|\varepsilon_{t-j}\right|) + \sum_{j=1}^{p} \beta_j \sigma_{t-j} \tag{1.26}$$

EWMA

A alternative to the series of GARCH models is the exponentially weighted moving average or EWMA model. This model calculates conditional variance based on the shocks from previous periods. The idea is that by including a smoothing parameter λ more weight is assigned to recent periods than distant periods. The λ must be less than 1. It is specified as in (1.27).

$$\sigma_t^2 = (1-\lambda)\sum_{j=1}^{\infty}(\lambda^j \varepsilon_{t-j}^2) \tag{1.27}$$

In practice a λ of 0.94 is often used, such as by the financial risk management company RiskMetrics^{TM} model of J.P. Morgan (Morgan Guaranty Trust Company 1996).

1.3 ACD models

An extension to GARCH models was proposed by Hansen (1994), the autoregressive conditional density estimation model (referred to as ACD models, sometimes ARCD). It focuses on time variation in higher moments (skewness and kurtosis), because the degree and frequency of extreme events seem to be not expected by traditional models. Some GARCH models are already able to capture the dynamics by relying on a different unconditional distribution than the normal distribution (for example skewed distributions like the SGED, SGT), or a model that allows to model these higher moments. However, Ghalanos (2016) mentions that these models also assume the shape and skewness parameters to be constant (not time varying). As Ghalanos mentions: “the research on time varying higher moments has mostly explored different parameterizations in terms of dynamics and distributions with little attention to the performance of the models out-of-sample and ability to outperform a GARCH model with respect to VaR.” Also one could question the marginal benefits of the ACD, while the estimation procedure is not simple (nonlinear bounding specification of higher moment distribution parameters and interaction). So, are skewness (skewness parameter) and kurtosis (shape parameters) time varying? The literature investigating higher moments has arguments for and against this statement. In part 2.2.2 the specification is given.

1.4 Value at Risk

Value-at-Risk (VaR) is a risk metric developed simultaniously by [Markowitz1952] and [Roy1952] to calculate how much money an investment, portfolio, department or institution such as a bank could lose in a market downturn, though in this period it remained mostly a theoretical discussion due to lacking processing power and industry demand for risk management measures. According to [Holton2002] VaR gained traction in the last decade of the 20th century when financial institutions started using it to determine their regulatory capital requirements. A VaR_{99} finds the amount that would be the greatest possible loss in 99% of cases. It can be defined as the threshold value θ_t. Put differently, in 1% of cases the loss would be greater than this amount.It is specified as in (1.28). [Christofferson2001] puts forth a general framework for specifying VaR models and comparing between two alternatives models.

$$Pr(y_t \leq \theta_t|\Omega_{t-1}) \equiv \phi \tag{1.28}$$

With y_t expected returns in period t, Ω_{t-1} the information set available in the previous period and ϕ the chosen confidence level.

1.5 Conditional Value at Risk

One major shortcoming of the VaR is that it does not provide information on the probability distribution of losses beyond the threshold amount. As VaR lacks subadditivity of different percentile outcomes, [Artzner1998] reject it as a coherent measure of risk. This is problematic, as losses beyond this amount would be more problematic if there is a large probability distribution of extreme losses, than if losses follow say a normal distribution. To solve this issue, they provide a conceptual idea of a conditional VaR (cVaR) which quantifies the average loss one would expect if the threshold is breached, thereby taking the distribution of the tail into account. Mathematically, a $cVaR_{99}$ is the average of all the VaR with a confidence level equal to or higher than 99. It is commonly referred to as expected shortfall (ES) sometimes and was written out in the form it is used by today by (Bertsimas et al. 2004). It is specified as in (1.29).

To calculate θ_t, VaR and cVaR require information on the expected distribution mean, variance and other parameters, to be calculated using the previously discussed GARCH models and distributions.

$$Pr(y_t \leq \theta_t|\Omega_{t-1}) \equiv \int_{-\infty}^{\theta_t} f(y_t|\Omega_{t-1})\,\mathrm{d}y_t = \phi \tag{1.29}$$

With the same notations as before, and f the (conditional) probability density function of y_t.

According to the BIS framework, banks need to calculate both VaR_{99} and $VaR_{97.5}$ daily to determine capital requirements for equity, using a minimum of one year of daily observations (Basel Committee on Banking Supervision 2016). Whenever a daily loss is recorded, this has to be registered as an exception. Banks can use an internal model to calculate their VaRs, but if they have more than 12 exceptions for their VaR_{99} or 30 exceptions for their $VaR_{97.5}$ they have to follow a standardized approach.Similarly, banks must calculate $cVaR_{97.5}$.

1.6 Past literature on the consequences of higher moments for VaR determination

Here comes the discussion about studies that have looked at higher moments and VaR determination. Also a summary of studies that discusses time-varying higher moments, but not a big part, while it is also only a small part of the empirical findings (couple of GARCH-ACD models).

Table 1.2: Higher moments and VaR

Author	Higher moments
Hansen (1994)	
\@harvey1999	
Brooks et al. (2005)	

2 Data and methodology

2.1 Data

We worked with daily returns on the EURO STOXX 50 Index denoted in EUR from 31 December, 1986 to 27 April, 2021. It is the leading blue-chip index of the Eurozone, was founded in 1999 and covers 50 of the most liquid and largest (in terms of free-float market capitalization) stocks. Its composition is reviewed annually in September, from each of the 19 EURO STOXX Supersector indices the biggest stocks are selected until the coverage is at 60% of the free-float market cap of each of the EURO STOXX Supersector index then all the current EURO STOXX 50 stocks are used in the selection list from which the largest 40 in terms of free-float market cap are selected and the remaining 10 stocks are chosen among those ranked between 41 and 60 (*Calculation guide STOXX* ® 2020).

The calculation of the index is made with the (2.1), that measures the changes in price of the index for fixed weights.

$$\text{Index}_{\,t} = \frac{\sum_{i=1}^{n}\left(p_{it} \cdot s_{it} \cdot ff_{it} \cdot cf_{it} \cdot x_{it}\right)}{D_t} = \frac{M_t}{D_t} \tag{2.1}$$

where: t = Time the index is computed n = Number of companies in the index p_{it} = Price of company (i) at time (t) s_{it} = Number of shares of company (i) at time (t) ff_{it} = Free float factor of company (i) at time (t) cf_{it} = Weighting cap factor of company (i) at time (t) x_{it} = Exchange rate from local currency into index currency for company (i) at time (t) M_t = Free-float market capitalization of the index at time (t) D_t = Divisor of the index at time (t)

Changes in weights caused by corporate actions are proportionally distributed across the components of the index and the index Divisor is computed with the (2.2) formula.

$$D_{t+1} = D_t \cdot \frac{\sum_{i=1}^{n} (p_{it} \cdot s_{it} \cdot ff_{it} \cdot cf_{it} \cdot x_{it}) \pm \Delta MC_{t+1}}{\sum_{i=1}^{n} (p_{it} \cdot s_{it} \cdot ff_{it} \cdot cf_{it} \cdot x_{it})} \tag{2.2}$$

where: ΔMC_{t+1} = Difference between the closing market capitalization of the index and the adjusted closing market capitalization of the index

(Optional)

The same analysis has been performed for the INDEX 1, INDEX 2, INDEX 3 and the INDEX 4 indexes with not different conclusions. The findings of these researches are available upon requests.

2.1.1 Descriptives

Table of summary statistics

Equation 2.1 provides the main statistics describing the return series analyzed. Returns are computed with equation (2.3).

$$R_t = 100\left(\ln\left(I_t\right) - \ln\left(I_{t-1}\right)\right) \tag{2.3}$$

where I_t is the index price at time t and I_{t-1} is the index price at $t-1$. The arithmetic mean of the series is 0.017% with a standard deviation of 1.307% and a median of 0.036 which translate to an annualized mean of 4.208% and an annualized standard deviation of 20.748%. The skewness statistic is highly significant and negative at -0.31 and the excess kurtosis is also highly significant and positive at 7.208. These 2 statistics give an overview of the distribution of the returns which has thicker tails than the normal distribution with a higher presence of left tail observations. A formal test such as the Jarque-Bera one with its statistic at 19528.62 and a high statistical significance, confirms the non normality feeling given by the Skewness and Kurtosis.

The right column of table 2.1 displays the same descriptive statistics but for the standardizes residuals obtained from a simple GARCH model as mentioned in table 2.1 in Note 2$*$. Again, Skewness statistic at -0.633 with a high statistical significance level and the excess Kurtosis at 5.134 also with a high statistical significance, suggest a non normal distribution of the standardized residuals and the Jarque-Bera statistic at NA, given its high significance, confirms the rejection of the normality assumption.

Table 2.1: Summary statistics of the returns

Statistics	Eurostoxx.50	Standardized.Residuals
Arithmetic Mean	0.0167	-0.0409
Median	0.0357	-0.0193
Maximum	10.4376	5.7126
Minimum	-13.2404	-11.7732
Stdev	1.307	0.9992
Skewness	-0.31	-0.6327
	(0***)	(0***)
Excess Kurtosis	7.2083	5.134
	(0***)	(0***)
Jarque-Bera	19528.6196***	10431.0514***

Notes

[1] This table shows the descriptive statistics of the daily percentage returns of EURO STOXX 50 over the period 1987-01-01 to 2021-04-27 (8954 observations). Including arithmetic mean, median, maximum, minimum, standard deviation, skewness, excess kurtosis and the Jarque-Bera test.

[2] The standardized residual is derived from a maximum likelyhood estimation (simple GARCH model) as follows:
$R_t = \alpha_0 + \alpha_1 R_{t-1} + z_t \sigma_t$
$\sigma_t^2 = \beta_0 + \beta_1 \sigma_{t-1}^2 z_{t-1}^2 + \beta_2 \sigma_{t-1}^2$,
Where z is the standard residual (assumed to have a normal distribution).

[3] *, **, *** represent significance levels at the 5

Descriptive figures

Stylized facts

As can be seen in figure 2.1 the Euro area equity and later, since 1999 the EuroStoxx 50 went up during the tech (“dot com”) bubble reaching an ATH of €5464.43. Then, there was a correction to boom again until the burst of the 2008 financial crisis. After which it decreased significantly. With an ATL at 09 March, 2009 of €1809.98. There is an improvement, but then the European debt crisis, with it’s peak in 2010-2012, occurred. From then there was some improvement until the “health crisis”, which arrived in Europe, February 2020. This crisis recovered very quickly reaching already values higher then the pre-COVID crisis level.

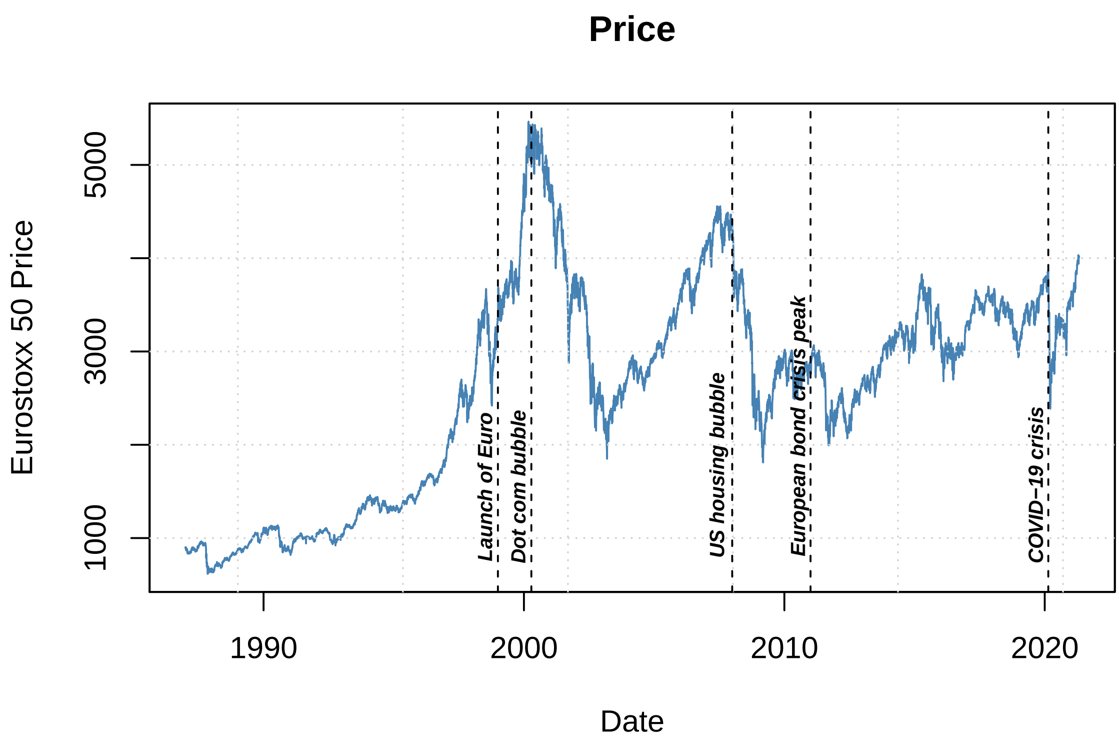

Figure 2.1: Eurostoxx 50 Price Index prices

In figure 2.2 the daily log-returns are visualized. A stylized fact that is observable is the volatility clustering. As can be seen: periods of large volatility are mostly followed by large volatility and small volatility by small volatility.

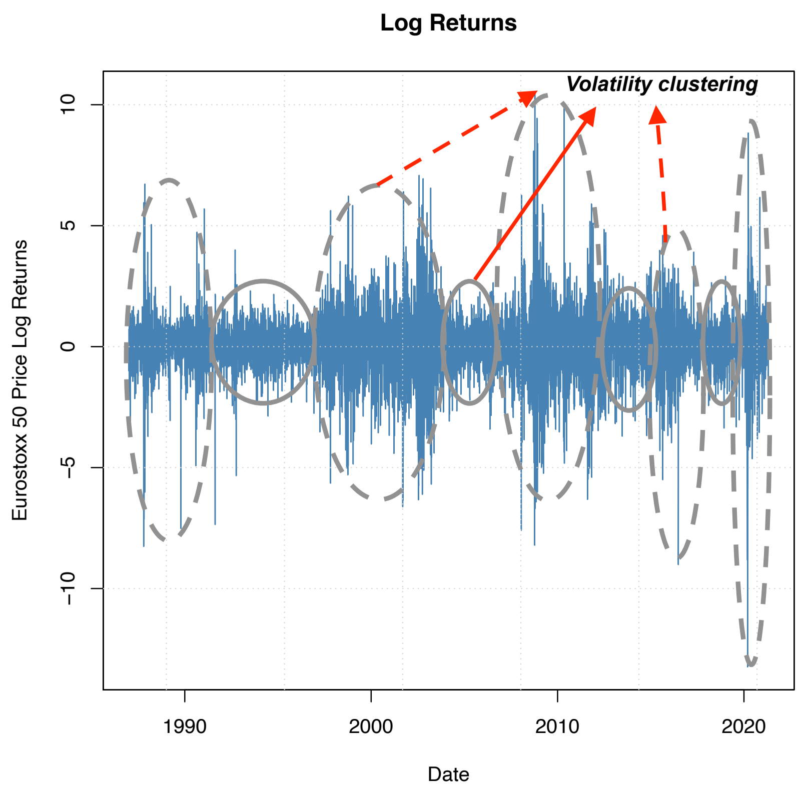

Figure 2.2: Eurostoxx 50 Price Index log returns

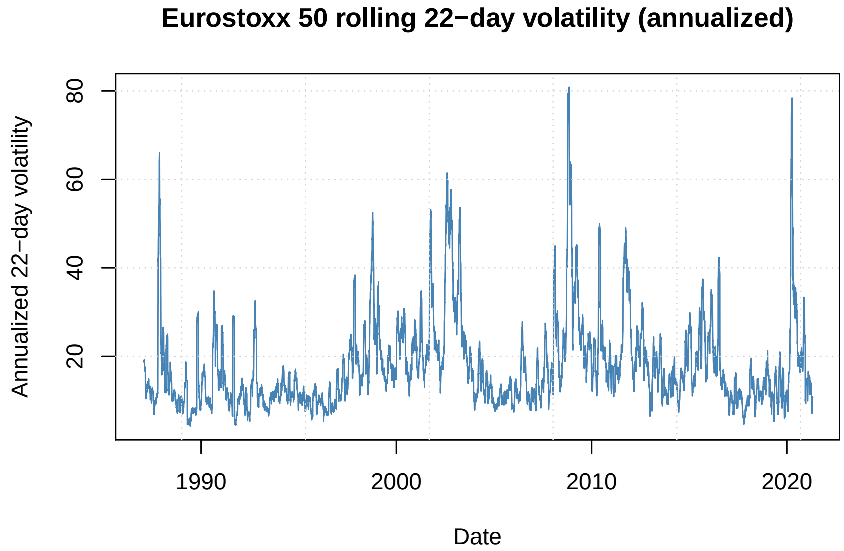

Figure 2.3: Eurostoxx 50 rolling volatility (22 days, calculated over 252 days)

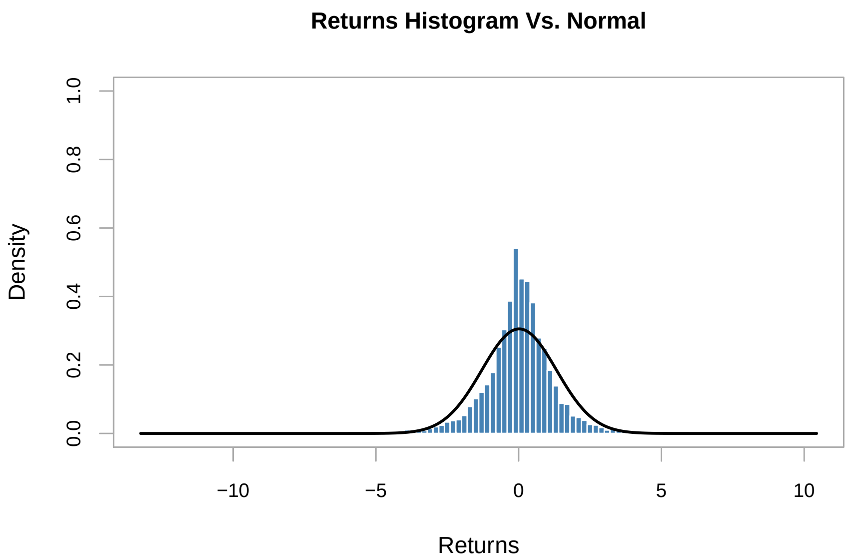

Figure 2.4: Density vs. Normal Eurostoxx 50 log returns)

In figure 2.4 the density distribution of the log returns are examined. As can be seen, as already mentioned in part 1.1, log returns are not really normally distributed. So

ACF plots: to do...

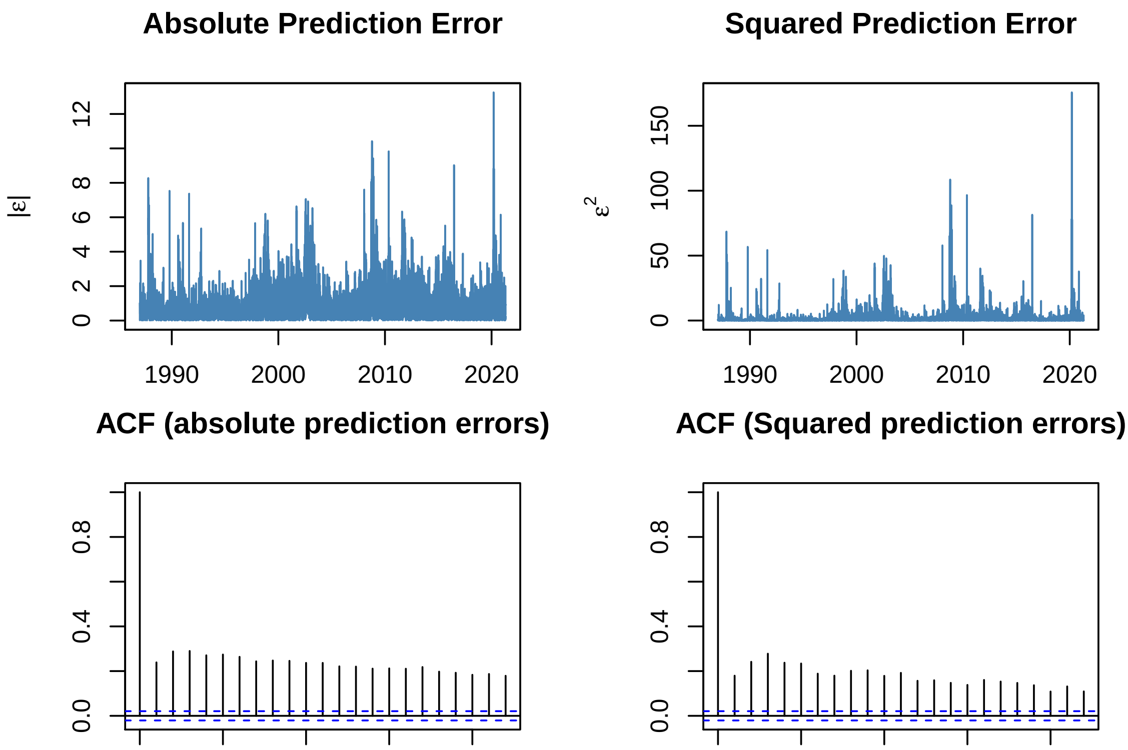

Figure 2.5: Absolute prediction errors

2.2 Methodology

2.2.1 Garch models

As already mentioned in part @(ref:univ-garch), GARCH models GARCH, EGARCH, IGARCH, GJRGARCH, NGARCH, TGARCH and NAGARCH (or TSGARCH) will be estimated. Additionally the distributions will be examined as well, including the normal, student-t distribution, skewed student-t distribution, generalized error distribution, skewed generalized error distribution and the skewed generalized t distribution.

They will be estimated using maximum likelihood. As already mentioned, fortunately, Ghalanos (2020b) has made it easy for us to implement this methodology in the R language (v.3.6.1) with the package "rugarch" v.1.4-4 (*R univariate garch*), which gives us a bit more time to focus on the results and the interpretation.

Maximum likelihood estimation is a method to find the distribution parameters that best fit the observed data, through maximization of the likelihood function, or the computationally more efficient log-likelihood function (by taking the natural logarithm). It is assumed that the return data is i.i.d. and that there is some underlying parametrized density function f with one or more parameters that generate the data, defined as a vector θ (equation (2.5)). These functions are based on the joint probability distribution of the observed data (equation (2.7)). Subsequently, the (log)likelihood function is maximized using an optimization algorithm (equation (2.9)).

$$y_1, y_2, ..., y_N \sim i.i.d \tag{2.4}$$

$$y_i \sim f(y|\theta) \tag{2.5}$$

$$L(\theta) = \prod_{i=1}^{N} f(y_i|\theta) \tag{2.6}$$

$$\log(L(\theta)) = \sum_{i=1}^{N} \log f(y_i|\theta) \tag{2.7}$$

$$\theta^* = arg \max_{\theta}[L] \tag{2.8}$$

$$\theta^* = arg \max_{\theta}[\log(L)] \tag{2.9}$$

2.2.2 ACD models

Following Ghalanos (2016), arguments of ACD models are specified as in Hansen (1994). The density function $f(y|\alpha)$ has parameters $\alpha_t = (\mu_t, \sigma_t, \nu_t)$, with equation (2.10), the conditional mean equation. Equation (2.11) as the conditional variance. And $\nu_t = \nu(\theta, x_t)$ the remaining parameters of the distribution like the skewness and kurtosis (shape) parameters.

$$\mu_t = \mu\left(\theta, x_t\right) = E\left(y_t \mid x_t\right) \tag{2.10}$$

$$\sigma_t^2 = \sigma^2\left(\theta, x_t\right) = E\left(\left(y_t - \mu_t^2\right) \mid x_t\right) \tag{2.11}$$

To further explain the difference between GARCH and ACD. The scaled innovations are given by equation (2.12). The conditional density is given by equation (2.13) and related to the density function $f(y|\alpha)$ as in equation (2.2.2).

$$z_t(\theta) = \frac{y_t - \mu\left(\theta, x_t\right)}{\sigma\left(\theta, x_t\right)} \tag{2.12}$$

$$g\left(z \mid \eta_t\right) = \frac{d}{dz} P\left(z_t < z \mid \eta_t\right) \tag{2.13}$$

$$f\left(y_t \mid \mu_t, \sigma_t^2, \eta_t\right) = \frac{1}{\sigma_t} g\left(z_t \mid \eta_t\right) \tag{2.14}$$

Again Ghalanos (2016) makes it easier to implement the somewhat complex ACD models using the R language with package "racd".

2.2.3 Control Tests

Unconditional coverage test of Kupiec (1995)

A number of tests are computed to see if the value-at-risk estimations capture the actual losses well. A first one is the unconditional coverage test by Kupiec (1995). The unconditional coverage or proportion of failures method tests if the actual value-at-risk exceedances are consistent with the expected exceedances (a chosen percentile, e.g. 1% percentile) of the VaR model. Following Kupiec (1995) and Ghalanos (2020a), the number of exceedence follow a binomial distribution (with thus probability equal to the significance level or expected proportion) under the null hypothesis of a correct VaR model. The test is conducted as a likelihood ratio test with statistic like in equation (2.15), with p the probability of an exceedence for a confidence level, N the sample size and X the number of exceedence. The null hypothesis states that the test statistic LR^{uc} is χ^2-distributed with one degree of freedom or that the probability of failure $\hat{p}$ is equal to the chosen percentile α.

$$LR^{uc} = -2\ln\left(\frac{(1-p)^{N-X}p^X}{\left(1-\frac{X}{N}\right)^{N-X}\left(\frac{X}{N}\right)^X}\right) \tag{2.15}$$

Conditional coverage test of Christoffersen et al. (2001)

Christoffersen et al. (2001) proposed the conditional coverage test. It is tests for unconditional covrage and serial independence. The serial independence is important while the LR^{uc} can give a false picture while at any point in time it classifies inaccurate VaR estimates as “acceptably accurate” (Bali and Theodossiou 2007). For a certain VaR estimate an indicator variable, $I_t(\alpha)$, is computed as equation (2.16).

$$I_t(\alpha) = \begin{cases} 1 & \text{if exceedence occurs} \\ 0 & \text{if no exceedence occurs} \end{cases} . \tag{2.16}$$

It involves a likelihood ratio test’s null hypothesis is that the statistic is χ^2-distributed with two degrees of freedom or that the probability of violation $\hat{p}$ (unconditional coverage) as well as the conditional coverage (independence) is equal to the chosen percentile α.

Dynamic quantile test

Engle and Manganelli (1999) with the aim to provide completeness to the conditional coverage test of Christoffersen et al. (2001) developed the Dynamic quantile test. It consists in testing some restriction in a . . . (work-in-progress).

3

Empirical Findings

3.1 Density of the returns

3.1.1 MLE distribution parameters

```
par(mfrow = c(3,2))
sgt.mle2 <- function (X.f, mu.f = mu ~ mu, sigma.f = sigma ~ sigma, lambda.f = l
  { formList = list(X = X.f, mu = mu.f, sigma = sigma.f, lambda = lambda.f,
                  p = p.f, q = q.f)
  varNames = NULL
  envir = new.env()
  for (i in 1:6) {
    formList[[i]] = stats::as.formula(formList[[i]])
    if (length(formList[[i]]) == 2L) {
      formList[[i]][[3L]] = formList[[i]][[2L]]
      formList[[i]][[2L]] = as.name(names(formList)[i])
    }
    else if (as.character(formList[[i]][[2L]]) != names(formList)[i]) {
      warning(paste("The left hand side of ", names(formList)[i],
                    ".f was changed from ", as.character(formList[[i]][[2L]]),
```

```
                    " to ", names(formList)[i], sep = ""))
  }
  varNames = c(varNames, all.vars(formList[[i]][[3L]]))
}
if (class(data)[1L] == "matrix")
  data = as.data.frame(data)
if (!is.list(data) && !is.environment(data))
  stop("'data' must be a list or an environment")
start = as.list(start)
if (is.null(names(start)))
  stop("'start' must be a named list or named numeric vector")
if ("" %in% names(start))
  stop("at least one of the elements in 'start' is missing a name")
parNames = names(start)
varNames = varNames[is.na(match(varNames, parNames))]
if (length(varNames) == 0L)
  stop("there is no reference to data in the given formulas")
for (i in varNames) {
  if (!exists(i, data))
    stop(paste(i, "is not contained in 'start' and it is not found in 'data'"))
  assign(i, eval(parse(text = paste("as.numeric(data$",
                                    i, ")", sep = ""))), envir)
}
if (length(varNames) > 1) {
  for (i in 2:length(varNames)) {
    if (length(eval(parse(text = paste("envir$", varNames[1L],
                                       sep = "")))) != length(eval(parse(text = pas

      stop(paste("the length of the variable", varNames[i],
                 "does not match the length of the variable",
```

```
                  varNames[1L]))
  }
}
control = list(...)
if (!is.null(control$maximize))
  stop("'maximize' option not allowed")
if (!missing(subset))
  for (i in varNames) assign(i, eval(parse(text = paste("envir$",
                                                        i, "[subset]", sep = "
keep = rep(TRUE, length(eval(parse(text = paste("envir$",
                                                varNames[1L], sep = "")))))
for (i in varNames) keep = keep & is.finite(eval(parse(text = paste("envir$",
                                                                    i, sep = "
for (i in varNames) assign(i, eval(parse(text = paste("envir$",
                                                      i, "[keep]", sep = "")))
loglik = function(params) {
  for (i in 1:length(parNames)) assign(parNames[i], unlist(params[i]))
  X = eval(formList[[1L]][[3L]])
  mu = eval(formList[[2L]][[3L]])
  sigma = eval(formList[[3L]][[3L]])
  lambda = eval(formList[[4L]][[3L]])
  p = eval(formList[[5L]][[3L]])
  q = eval(formList[[6L]][[3L]])
  sum(dsgt(X, mu, sigma, lambda, p, q, mean.cent, var.adj,
           log = TRUE))
}
environment(loglik) = envir
negloglik = function(params) {
  -loglik(params)
}
```

```
if (!is.finite(loglik(start)))
  stop("'start' yields infinite or non-computable SGT function values")
optimum = suppressWarnings(optimx::optimx(par = unlist(start),
                                          fn = negloglik, method = method, itnmax =
minimum = min(optimum$value, na.rm = TRUE)
if (!is.finite(minimum))
  stop("All Maximization Methods Failed")
whichbest = max(which(minimum == optimum$value))
optimal = optimum[whichbest, ]
estimate = as.numeric(optimum[whichbest, 1:length(parNames)])
names(estimate) = parNames
H = tryCatch(numDeriv::hessian(loglik, estimate, method = hessian.method),
             error = function(e) {
               warning("hessian matrix calculation failed")
               return(as.matrix(NaN))
             })
varcov = tryCatch(-qr.solve(H), error = function(e) {
  warning("covariance matrix calculation failed due to a problem with the hessian")
  return(as.matrix(NaN))
})
std.error = sqrt(diag(varcov))
if (is.finite(varcov[1, 1]))
  names(std.error) = parNames
gradient = tryCatch(numDeriv::grad(loglik, estimate, method = gradient.method),
                    error = function(e) {
                      warning("gradient calculation failed")
                      return(NaN)
                    })
result = list(maximum = -minimum, estimate = estimate, convcode = as.numeric(optima
              niter = as.numeric(optimal$niter), best.method.used = row.names(optim
```

```
                optimx = optimum, hessian = H, gradient = gradient,
                varcov = varcov, std.error = std.error)
  class(result) = c("sgtest", class(result))
  return(result)
}
```

```
library(sgt)
require(graphics)
require(stats)



DistMLE <- function(series) {

  ### SGT
  X.data <- X ~ coredata(series)

  SGT_start <- list(mu = 0, sigma = 2, lambda = 0, p = 2, q = 12)
  SGT_result <- sgt.mle2(X.f = X.data, start = SGT_start)
  SGT_sumResult <- summary(SGT_result)
  SGT_AIC <- 2*length(SGT_result$estimate) - 2*SGT_sumResult$maximum

  ### SGT plot fit
  xvals = seq(-3,3,by=0.01)
  SGT_mu <- SGT_result$estimate[1]
  SGT_sigma <- SGT_result$estimate[2]
  SGT_lambda <- SGT_result$estimate[3]
  SGT_p <- SGT_result$estimate[4]
  SGT_q <- SGT_result$estimate[5]
  plot(xvals, dgst(xvals, mu = SGT_mu, sigma = SGT_sigma, lambda = SGT_lambda, p
  lines(density(coredata(series)))
```

```
### SGED (sgt.mle2)

SGED_start <- list(mu = 0, sigma = 2, lambda = 0, p = 2, q = 750)
SGED_result <- sgt.mle2(X.f = X.data, start = SGED_start, lower = c(-Inf, -Inf, -In
SGED_sumResult <- summary(SGED_result)
SGED_AIC <- 2*length(SGED_result$estimate-1) - 2*SGED_sumResult$maximum

### SGED Plot fit (sgt.mle2)
SGED_mu <- SGED_result$estimate[1]
SGED_sigma <- SGED_result$estimate[2]
SGED_lambda <- SGED_result$estimate[3]
SGED_p <- SGED_result$estimate[4]
SGED_q <- SGED_result$estimate[5]
plot(xvals, dsgt(xvals, mu = SGED_mu, sigma = SGED_sigma, lambda = SGED_lambda, p =
lines(density(coredata(series)))

### GT(sgt.mle2)
# GT_start <- list(mu = 0, sigma = 2, lambda = 0, p = 2, q = 12)
# GT_result <- sgt.mle2(X.f = X.data, start = GT_start, lower = c(-Inf, -Inf, -0.
# GT_sumResult <- summary(GT_result)
# GT_AIC <- 2*length(GT_result$estimate-1) - 2*GT_sumResult$maximum

### GT Plot fit (sgt.mle2)
# GT_mu <- GT_result$estimate[1]
# GT_sigma <- GT_result$estimate[2]
# GT_lambda <- GT_result$estimate[3]
# GT_p <- GT_result$estimate[4]
```

```
# GT_q <- GT_result$estimate[5]
# plot(xvals, dsgt(xvals, mu = GT_mu, sigma = GT_sigma, lambda = GT_lambda, p
# lines(density(coredata(series)))

### GED(sgt.mle2)
GED_start <- list(mu = 0, sigma = 2, lambda = 0, p = 2, q = 750)
GED_result <- sgt.mle2(X.f = X.data, start = GED_start, lower = c(-Inf, -Inf, -
GED_sumResult <- summary(GED_result)
GED_AIC <- 2*length(GED_result$estimate-2) - 2*GED_sumResult$maximum

### GED Plot fit (sgt.mle2)

GED_mu <- GED_result$estimate[1]
GED_sigma <- GED_result$estimate[2]
GED_lambda <- GED_result$estimate[3]
GED_p <- GED_result$estimate[4]
GED_q <- GED_result$estimate[5]
plot(xvals, dsgt(xvals, mu = GED_mu, sigma = GED_sigma, lambda = GED_lambda, p
lines(density(coredata(series)))

### ST (fitdist)
ST_start <- list(mean=0,sd=2, nu = 8, xi=2)
ST_result <- fitdistrplus::fitdist(data = as.vector(coredata(R)), distr = "sstd
ST_sumResult <- summary(ST_result)
ST_sumResult$aic

### ST Plot fit (fitdist)
ST_mean <- ST_result$estimate[1]
ST_sd <- ST_result$estimate[2]
ST_nu <- ST_result$estimate[3]
```

```
ST_xi <- ST_result$estimate[4] #lamda

plot(xvals, dsstd(xvals, mean = ST_mean, sd = ST_sd, nu = ST_nu, xi=ST_xi), col="re
lines(density(coredata(series)))

### T (fitdist)
T_start <- list(mean = 0, sd = 1, nu = 5)
T_result <- fitdistrplus::fitdist(data = as.vector(coredata(R)), distr = "std", met
T_sumResult <- summary(T_result)

### T Plot fit (fitdist)
T_mean <- T_result$estimate[1]
T_sd <- T_result$estimate[2]
T_nu <- T_result$estimate[3]


plot(xvals, dstd(xvals, mean = T_mean, sd = T_sd, nu = T_nu), col="red", type ="l",
lines(density(coredata(series)))


### Normal (sgt.mle2)
Normal_start <- list(mu = 0, sigma = 2, lambda = 0, p = 2, q = 950)
Normal_result <- sgt.mle2(X.f = X.data, start = Normal_start, lower = c(-Inf, -Inf,
Normal_sumResult <- summary(Normal_result)
Normal_AIC <- 2*length(Normal_result$estimate-3) - 2*Normal_sumResult$maximum

### Normal  Plot fit (sgt.mle2)

Normal_mu <- Normal_result$estimate[1]
Normal_sigma <- Normal_result$estimate[2]
```

```
  Normal_lambda <- Normal_result$estimate[3]
  Normal_p <- Normal_result$estimate[4]
  Normal_q <- Normal_result$estimate[5]

  if(!is.na(Normal_mu)){
  plot(xvals, dsgt(xvals, mu = Normal_mu, sigma = Normal_sigma, lambda = Normal_l
 lines(density(coredata(series)))}

#maximum likelihood estimates of unconditional distribution functions

Table2 <- matrix(nrow = 6, ncol = 8)
colnames(Table2) <- c("mu","sigma","lambda","p","q","nu","L","AIC")
rownames(Table2) <- c("SGT","SGED","GED","ST","T","Normal")

Table2[1,1] <- SGT_mu
Table2[1,2] <- SGT_sigma
Table2[1,3] <- SGT_lambda
Table2[1,4] <- SGT_p
Table2[1,5] <- SGT_q
Table2[1,7] <- SGT_result$maximum
Table2[1,8] <- SGT_AIC

Table2[2,1] <- SGED_mu
Table2[2,2] <- SGED_sigma
Table2[2,3] <- SGED_lambda
Table2[2,4] <- SGED_p
Table2[2,5] <- SGED_q
Table2[2,7] <- SGED_result$maximum
Table2[2,8] <- SGT_AIC
```

```
# Table2[3,1] <- GT_mu
# Table2[3,2] <- GT_sigma
# Table2[3,3] <- GT_lambda
# Table2[3,4] <- GT_p
# Table2[3,5] <- GT_q
# Table2[3,7] <- GT_result$maximum
# Table2[3,8] <- GT_AIC



Table2[3,1] <- GED_mu
Table2[3,2] <- GED_sigma
Table2[3,3] <- GED_lambda
Table2[3,4] <- GED_p
Table2[3,5] <- GED_q
Table2[3,7] <- GED_result$maximum
Table2[3,8] <- GED_AIC

Table2[4,1] <- ST_mean
Table2[4,2] <- ST_sd
Table2[4,3] <- ST_xi
Table2[4,6] <- ST_nu
Table2[4,7] <- ST_result$loglik
Table2[4,8] <- ST_result$aic

Table2[5,1] <- T_mean
Table2[5,2] <- T_sd
Table2[5,6] <- T_nu
Table2[5,7] <- T_result$loglik
Table2[5,8] <- T_result$aic
```

```
if(!is.na(Normal_mu)){
Table2[6,1] <- Normal_mu
Table2[6,2] <- Normal_sigma
Table2[6,3] <- Normal_lambda
Table2[6,4] <- Normal_p
Table2[6,5] <- Normal_q
Table2[6,7] <- Normal_result$maximum
Table2[6,8] <- Normal_AIC
}

Table2
}
```

```
MLE_Eurostoxx <- DistMLE(R)
```

633 In table ?? we can see...

```
# MLE_Bali <- DistMLE(Rbali)
str(MLE_Eurostoxx)
table2 <- as.data.frame(MLE_Eurostoxx)
table2 <- round(table2,3)
table2[is.na(table2)] <- ""
table2 %>% kbl(caption = "Unconditional density distributions","latex",
      label = 'disttable',
      booktabs = T,
      position = "h!",
      digits = 3 )%>%
  kable_classic(full_width = F)%>%
  footnote(general = "Notes",number="Here comes text",threeparttable = T,footnote
```

3.2 Results of GARCH with constant higher moments

```
distributions <- c("norm", "std", "sstd", "ged", "sged")
#garchspec <- garchfit <- garchforecast <- stdret <- vector(mode = "list", length =
#names(garchspec) <- names(garchfit) <- names(garchforecast) <- names(stdret) <- di
Models.garch <- c("sGARCH", "eGARCH","fGARCH.AVGARCH","fGARCH.NAGARCH", "gjrGARCH", "

for(i in 1:length(Models.garch)){
assign(paste0("garchspec.",Models.garch[i]),vector(mode = "list", length = length(dis
assign(paste0("garchfit.",Models.garch[i]),vector(mode = "list", length = length(dist
assign(paste0("stdret.",Models.garch[i]),vector(mode = "list", length = length(distri
}


# ls(pattern = "garchspec.")
# sapply(ls(pattern = "garchspec."), FUN = setNames, distributions)


#.sGARCH--------------------------
for(i in 1:length(distributions)){
# Specify a GARCH model with constant mean
garchspec.sGARCH[[i]] <- ugarchspec(mean.model = list(armaOrder = c(1,0)),
                     variance.model = list(model = "sGARCH", garchOrder = c(1,1), var
                     distribution.model = distributions[i])
# Estimate the model
garchfit.sGARCH[[i]] <- ugarchfit(data = R, spec = garchspec.sGARCH[[i]])
# Compute stdret using residuals()
stdret.sGARCH[[i]] <- residuals(garchfit.sGARCH[[i]], standardize = TRUE)
}


#.eGARCH-------------------
```

```
for(i in 1:length(distributions)){
# Specify a GARCH model with constant mean
garchspec.eGARCH[[i]] <- ugarchspec(mean.model = list(armaOrder = c(1,0)),
                     variance.model = list(model = "eGARCH", variance.targeting =
                     distribution.model = distributions[i])
# Estimate the model
garchfit.eGARCH[[i]] <- ugarchfit(data = R, spec = garchspec.eGARCH[[i]])
# Compute stdret using residuals()
stdret.eGARCH[[i]] <- residuals(garchfit.eGARCH[[i]], standardize = TRUE)
}

#.fGARCH.NAGARCH------------------------
for(i in 1:length(distributions)){
# Specify a GARCH model with constant mean
garchspec.fGARCH.NAGARCH[[i]] <- ugarchspec(mean.model = list(armaOrder = c(1,0)
                     variance.model = list(model = "fGARCH", submodel = "NAGARCH
                     distribution.model = distributions[i])
# Estimate the model
garchfit.fGARCH.NAGARCH[[i]] <- ugarchfit(data = R, spec = garchspec.fGARCH.NAGA
# Compute stdret using residuals()
stdret.fGARCH.NAGARCH[[i]] <- residuals(garchfit.fGARCH.NAGARCH[[i]], standardiz
}

#.fGARCH.AVGARCH------------------------
for(i in 1:length(distributions)){
# Specify a GARCH model with constant mean
garchspec.fGARCH.AVGARCH[[i]] <- ugarchspec(mean.model = list(armaOrder = c(1,0)
                     variance.model = list(model = "fGARCH", submodel = "AVGARCH
                     distribution.model = distributions[i])
# Estimate the model
```

```
garchfit.fGARCH.AVGARCH[[i]] <- ugarchfit(data = R, spec = garchspec.fGARCH.AVGARCH[[
# Compute stdret using residuals()
stdret.fGARCH.AVGARCH[[i]] <- residuals(garchfit.fGARCH.AVGARCH[[i]], standardize = T
}

#.gjrGARCH------------------
for(i in 1:length(distributions)){
# Specify a GARCH model with constant mean
garchspec.gjrGARCH[[i]] <- ugarchspec(mean.model = list(armaOrder = c(1,0)),
                     variance.model = list(model = "gjrGARCH", variance.targeting = F
                     distribution.model = distributions[i])
# Estimate the model
garchfit.gjrGARCH[[i]] <- ugarchfit(data = R, spec = garchspec.gjrGARCH[[i]])
# Compute stdret using residuals()
stdret.gjrGARCH[[i]] <- residuals(garchfit.gjrGARCH[[i]], standardize = TRUE)
}

#fGARCH.TGARCH-------------------
for(i in 1:length(distributions)){
# Specify a GARCH model with constant mean
garchspec.fGARCH.TGARCH[[i]] <- ugarchspec(mean.model = list(armaOrder = c(1,0)),
                     variance.model = list(model = "fGARCH", submodel = "TGARCH", var
                     distribution.model = distributions[i])
# Estimate the model
garchfit.fGARCH.TGARCH[[i]] <- ugarchfit(data = R, spec = garchspec.fGARCH.TGARCH[[i]
# Compute stdret using residuals()
stdret.fGARCH.TGARCH[[i]] <- residuals(garchfit.fGARCH.TGARCH[[i]], standardize = TRU
}

#.iGARCH--------------------
```

```
for(i in 1:length(distributions)){
# Specify a GARCH model with constant mean
garchspec.iGARCH[[i]] <- ugarchspec(mean.model = list(armaOrder = c(1,0)),
                    variance.model = list(model = "iGARCH", variance.targeting =
                    distribution.model = distributions[i])
# Estimate the model
garchfit.iGARCH[[i]] <- ugarchfit(data = R, spec = garchspec.iGARCH[[i]])
# Compute stdret using residuals()
stdret.iGARCH[[i]] <- residuals(garchfit.iGARCH[[i]], standardize = TRUE)
}

#.csGARCH-----------------
# for(i in 1:length(distributions)){
# # Specify a GARCH model with constant mean
# garchspec.csGARCH[[i]] <- ugarchspec(mean.model = list(armaOrder = c(1,0)),
#                     variance.model = list(model = "csGARCH", variance.target
#                     distribution.model = distributions[i])
# # Estimate the model
# garchfit.csGARCH[[i]] <- ugarchfit(data = R, spec = garchspec.csGARCH[[i]])
# # Compute stdret using residuals()
# stdret.csGARCH[[i]] <- residuals(garchfit.csGARCH[[i]], standardize = TRUE)
# }



# we need EWMA
for(i in 1:length(distributions)){
# Specify a GARCH model with constant mean
garchspec.EWMA[[i]] <- ugarchspec(mean.model = list(armaOrder = c(1,0)),
                    variance.model = list(model = "iGARCH", variance.targeting =
                    distribution.model = distributions[i], fixed.pars = list(ome
```

```
# Estimate the model
garchfit.EWMA[[i]] <- ugarchfit(data = R, spec = garchspec.EWMA[[i]])
# Compute stdret using residuals()
stdret.EWMA[[i]] <- residuals(garchfit.EWMA[[i]], standardize = TRUE)
}



#  make the histogram
#
# chart.Histogram(stdret.iGARCH[[1]], methods = c("add.normal","add.density" ),
#                  colorset = c("gray","red","blue"))
```

```
table3 <- matrix(nrow = 12, ncol = 5)
colnames(table3) <- distributions

#trying a loop, maybe you can solve that @filippo?
## column loop i = normal distribution, std, sstd, ged, sged
table3[1,1] <- garchfit.sGARCH[[1]]@fit$coef[1] #first parameter estimate
table3[2,1] <- garchfit.sGARCH[[1]]@fit$se.coef[1] #first standard error
table3[3,1] <- garchfit.sGARCH[[1]]@fit$se.coef[2] #second parameter estimate
table3[4,1] <- garchfit.sGARCH[[1]]@fit$se.coef[2]
#...
table3 <- round(table3, 3)

# for (i in length(distributions)) {
#   for (j in nrow(table3)) {
#     table3[j,i] <- garchfit.sGARCH[[i]]@fit$coef
#     table3[j+1,i] <-garchfit.sGARCH[[i]]@fit$se.coef
#     }
# }
```

```
print("sGARCH")
garchfit.sGARCH[[1]]@fit$coef
garchfit.sGARCH[[1]]@fit$se.coef
```

```
print("iGARCH")
garchfit.iGARCH[[1]]@fit$coef
garchfit.iGARCH[[1]]@fit$se.coef
```

```
print("EWMA")
garchfit.EWMA[[1]]@fit$coef
c(garchfit.EWMA[[1]]@fit$se.coef[1:2],NA,garchfit.EWMA[[1]]@fit$se.coef[3], NA)
```

```
print("eGARCH")
garchfit.eGARCH[[1]]@fit$coef
garchfit.eGARCH[[1]]@fit$se.coef
```

```
print("gjrGARCH")
garchfit.gjrGARCH[[1]]@fit$coef
garchfit.gjrGARCH[[1]]@fit$se.coef
```

```
print("NAGARCH")
garchfit.fGARCH.NAGARCH[[1]]@fit$coef
garchfit.fGARCH.NAGARCH[[1]]@fit$se.coef
```

```
print("TGARCH")
garchfit.fGARCH.TGARCH[[1]]@fit$coef
garchfit.fGARCH.TGARCH[[1]]@fit$se.coef
```

```
print("TSGARCH/AVGARCH")
garchfit.fGARCH.AVGARCH[[1]]@fit$coef
garchfit.fGARCH.AVGARCH[[1]]@fit$se.coef
```

636 ## 3.3 Results of GARCH with time-varying higher
637 moments

```
require(racd)
require(rugarch)
require(parallel)
require(xts)
# ACD specification
sGARCH_ACDspec = acdspec(mean.model = list(armaOrder = c(1, 0)), variance.model = lis
distribution.model = list(model = 'jsu', skewOrder = c(1, 1, 1), shapeOrder = c(1,1,1

# sGARCH
cl = makePSOCKcluster(10)
fit = acdfit(sGARCH_ACDspec, as.data.frame(R), solver = 'msoptim', solver.control = l
```

```
# plotxts comes from implementing https://stackoverflow.com/a/50051183/271616
# par(mfrow = c(2, 2), mai = c(0.75, 0.75, 0.3, 0.3))
# cm <- plot.zoo(xts(fit@model$modeldata$data, fit@model$modeldata$index), auto.gri
# cm <- lines(fitted(fit), col = 2)
# cm
# cs <- plot(xts(abs(fit@model$modeldata$data),fit@model$modeldata$index), auto.gri
# minor.ticks = FALSE, main = 'Conditional Sigma', yaxis.right = F,col = 'grey')
# cs <- lines(sigma(fit), col = 'steelblue')
# cs
# plot(racd::skewness(fit), col = 'steelblue',yaxis.right = F, main = 'Conditional
# plot(racd::kurtosis(fit), col = 'steelblue', yaxis.right = F,main = 'Conditional 

# pnl <- function(fitted(fit),xts(fit@model$modeldata$data, fit@model$modeldata$ind
#   panel.number <- parent.frame()$panel.number
#   if (panel.number == 1) lines(fitted(fit), xts(fit@model$modeldata$data, fit@mod
```

```
#   lines(fitted(fit),xts(fit@model$modeldata$data, fit@model$modeldata$index),
# }
# plot(xts(fit@model$modeldata$data, fit@model$modeldata$index), auto.grid = T,
# # lines(fitted(fit), col = 2) + grid()
#
# plot(xts(fit@model$modeldata$data, fit@model$modeldata$index), auto.grid = T,
```

4

Robustness Analysis

4.1 Specification checks

In order to check if the models are specified correctly, some specification checks have to be performed. The specification checks have to be done on the standardized residuals of the estimated GARCH model given by the following equation:

$$\hat{Z}_t = \frac{\hat{\varepsilon}_t}{\hat{\sigma}_t} = \frac{R_t - \hat{\mu}}{\hat{\sigma}_t}$$

4.1.1 Figures control tests

Autocorrelation function of the standardized residuals and autocorrelation function of the squared standardized residuals.

Then the density can be examined standardized residuals and compared with the normal distribution.

Also the QQ-plot can be examined.

4.1.2 Residual heteroscedasticity

Ljung-Box test on the squared or absolute standardized residuals.

zero-mean unit-variance not skewed no excess kurtosis no serial correlation in the squares no serial correlation in the cubes no serial correlation in the squares

5 Conclusion

Appendices

A
Appendix

Works Cited

Annaert, Jan (Jan. 2021). *Quantitative Methods in Finance*. Version 0.2.1. Antwerp Management School.

Bali, Turan G., Hengyong Mo, and Yi Tang (Feb. 2008). "The role of autoregressive conditional skewness and kurtosis in the estimation of conditional VaR". In: *Journal of Banking and Finance* 32.2. Publisher: North-Holland, pp. 269–282. DOI: `10.1016/j.jbankfin.2007.03.009`.

Bali, Turan G. and Panayiotis Theodossiou (Feb. 22, 2007). "A conditional-SGT-VaR approach with alternative GARCH models". In: *Annals of Operations Research* 151.1, pp. 241–267. DOI: `10.1007/s10479-006-0118-4`. URL: `http://link.springer.com/10.1007/s10479-006-0118-4`.

Basel Committee on Banking Supervision (2016). *Minimum capital requirements for market risk*. Tech. rep. Issue: January Publication Title: Bank for International Settlements, pp. 92–92. URL: `https://www.bis.org/basel_framework/`.

Bertsimas, Dimitris, Geoorey J Lauprete, and Alexander Samarov (2004). "Shortfall as a risk measure: properties, optimization and applications". In: *Journal of Economic Dynamics and Control* 28, pp. 1353–1381. DOI: `10.1016/S0165-1889(03)00109-X`.

Bollerslev, Tim (1986). "Generalized Autoregressive Conditional Heteroskedasticity". In: *Journal of Econometrics* 31, pp. 307–327.

— (1987). "A Conditionally Heteroskedastic Time Series Model for Speculative Prices and Rates of Return". In: *The Review of Economics and Statistics* 69.3. Publisher: The MIT Press, pp. 542–547. DOI: `10.2307/1925546`. URL: `https://www.jstor.org/stable/1925546`.

— (Sept. 4, 2008). "Glossary to ARCH (GARCH)". In: p. 46. DOI: `10.2139/ssrn.1263250`. URL: `https://ssrn.com/abstract=1263250`.

Bollerslev, Tim, Robert F. Engle, and Daniel B. Nelson (Jan. 1994). "Chapter 49 Arch models". In: *Handbook of Econometrics* 4. Publisher: Elsevier, pp. 2959–3038. DOI: `10.1016/S1573-4412(05)80018-2`.

Brooks, Chris et al. (2005). "Autoregressive conditional kurtosis". In: *Journal of Financial Econometrics* 3.3, pp. 399–421. DOI: `10.1093/jjfinec/nbi018`.

Calculation guide STOXX ® (2020). Tech. rep.

Christoffersen, Peter, Jinyong Hahn, and Atsushi Inoue (July 2001). "Testing and comparing Value-at-Risk measures". In: *Journal of Empirical Finance* 8.3, pp. 325–342. DOI: `10.1016/S0927-5398(01)00025-1`.

Davidian, M. and R. J. Carroll (Dec. 1987). "Variance Function Estimation". In: *Journal of the American Statistical Association* 82.400. Publisher: JSTOR, pp. 1079–1079. DOI: `10.2307/2289384`.

Engle, R. F. (1982). "Autoregressive Conditional Heteroscedacity with Estimates of variance of United Kingdom Inflation,journal of Econometrica, Volume 50, Issue 4 (Jul., 1982),987-1008." In: *Econometrica* 50.4, pp. 987–1008.

Engle, Robert (2001). "GARCH 101: The use of ARCH/GARCH models in applied econometrics". In: *Journal of Economic Perspectives*. DOI: 10.1257/jep.15.4.157.

Engle, Robert F. and S. Manganelli (1999). *CAViaR: Conditional Autoregressive Value at Risk by Regression Quantiles*. Tech. rep. San Diego: UC San Diego. URL: http://www.jstor.org/stable/1392044.

Engle, Robert F. and Victor K. Ng (Dec. 1993). "Measuring and Testing the Impact of News on Volatility". In: *The Journal of Finance* 48.5. Publisher: John Wiley and Sons, Ltd, pp. 1749–1778. DOI: 10.1111/j.1540-6261.1993.tb05127.x.

Fama, Eugene (1970). *Efficient Capital Markets: A Review of Theory and Empirical Work*. Tech. rep. 2, pp. 383–417. DOI: 10.2307/2325486.

Fama, Eugene F. (1965). "The Behavior of Stock-Market Prices". In: *The Journal of Business* 38.1, pp. 34–105. URL: http://www.jstor.org/stable/2350752.

Fernández, Carmen and Mark F. J. Steel (Mar. 1998). "On Bayesian Modeling of Fat Tails and Skewness". In: *Journal of the American Statistical Association* 93.441, pp. 359–371. DOI: 10.1080/01621459.1998.10474117. URL: http://www.tandfonline.com/doi/abs/10.1080/01621459.1998.10474117.

Ghalanos, Alexios (2016). *racd: Autoregressive Conditional Density Models*. http://www.unstarched.net, https://bitbucket.org/alexiosg/.

— (2020a). *Introduction to the rugarch package. (Version 1.4-3)*. URL: http://cran.r-project.org/web/packages/rugarch/.

— (2020b). *rugarch: Univariate GARCH models*. R package version 1.4-4.

Giot, Pierre and Sébastien Laurent (Nov. 2003). "Value-at-risk for long and short trading positions". In: *Journal of Applied Econometrics* 18.6, pp. 641–663. DOI: 10.1002/jae.710. URL: http://doi.wiley.com/10.1002/jae.710.

— (June 1, 2004). "Modelling daily Value-at-Risk using realized volatility and ARCH type models". In: *Journal of Empirical Finance* 11.3, pp. 379–398. DOI: 10.1016/j.jempfin.2003.04.003. URL: https://www.sciencedirect.com/science/article/pii/S092753980400012X.

Glosten, Lawrence R., Ravi Jagannathan, and David E. Runkle (Dec. 1993). "On the Relation between the Expected Value and the Volatility of the Nominal Excess Return on Stocks". In: *The Journal of Finance* 48.5. Publisher: John Wiley and Sons, Ltd, pp. 1779–1801. DOI: 10.1111/j.1540-6261.1993.tb05128.x. URL: http://doi.wiley.com/10.1111/j.1540-6261.1993.tb05128.x.

Hansen, Bruce E. (1994). "Autoregressive Conditional Density Estimation". In: *International Economic Review* 35.3, pp. 705–730.

Jorion, Philippe (2007). *Value at Risk: The New Benchmark For Managing Financial Risk*. 3rd ed. McGraw-Hill.

Kupiec, P.H. (1995). "Techniques for Verifying the Accuracy of Risk Measurement Models". In: *Journal of Derivatives* 3.2, pp. 73–84. DOI: 10.3905/jod.1995.407942.

Lee, Ming-Chih, Jung-Bin Su, and Hung-Chun Liu (Oct. 22, 2008). "Value-at-risk in US stock indices with skewed generalized error distribution". In: *Applied Financial Economics Letters* 4.6, pp. 425–431. DOI: 10.1080/17446540701765274. URL: http://www.tandfonline.com/doi/abs/10.1080/17446540701765274.

Lyngs, Ulrik (2019). *oxforddown: An Oxford University Thesis Template for R Markdown*. https://github.com/ulyngs/oxforddown. DOI: 10.5281/zenodo.3484682.

McDonald, James B. and Whitney K. Newey (Dec. 1988). "Partially Adaptive Estimation of Regression Models via the Generalized *T* Distribution". In: *Econometric Theory*

4.3, pp. 428–457. DOI: 10.1017/S0266466600013384. URL: https://www.cambridge.org/core/product/identifier/S0266466600013384/type/journal_article.

Morgan Guaranty Trust Company (1996). *RiskMetricsTM—Technical Document.* Tech. rep.

Nelson, Daniel B. (Mar. 1991). "Conditional Heteroskedasticity in Asset Returns: A New Approach". In: *Econometrica* 59.2. Publisher: JSTOR, pp. 347–347. DOI: 10.2307/2938260.

Officer, R. R. (1972). *The Distribution of Stock Returns.* Tech. rep. 340, pp. 807–812.

Schwert, G. William (1989). "Why Does Stock Market Volatility Change Over Time?" In: *The Journal of Finance* 44.5, pp. 1115–1153. DOI: 10.1111/j.1540-6261.1989.tb02647.x.

Taylor, Stephen J. (1986). *Modelling financial time series.* Chichester: John Wiley and Sons, Ltd.

Theodossiou, Panayiotis (1998). "Financial data and the skewed generalized t distribution". In: *Management Science* 44.12 part 1. Publisher: INFORMS Inst.for Operations Res.and the Management Sciences, pp. 1650–1661. DOI: 10.1287/mnsc.44.12.1650.

Theodossiou, Peter (2000). "Skewed Generalized Error Distribution of Financial Assets and Option Pricing". In: *SSRN Electronic Journal.* DOI: 10.2139/ssrn.219679. URL: http://www.ssrn.com/abstract=219679.

Trottier, Denis-Alexandre and David Ardia (Sept. 4, 2015). "Moments of standardized Fernandez-Steel skewed distributions: Applications to the estimation of GARCH-type models". In: *Finance Research Letters* 18, pp. 311–316. DOI: 10.2139/ssrn.2656377. URL: https://ssrn.com/abstract=2656377.

Welch, Ivo and Amit Goyal (July 2008). "A Comprehensive Look at The Empirical Performance of Equity Premium Prediction". In: *Review of Financial Studies* 21.4, pp. 1455–1508. DOI: 10.1093/rfs/hhm014. URL: https://academic.oup.com/rfs/article-lookup/doi/10.1093/rfs/hhm014.

Zakoian, Jean Michel (1994). "Threshold heteroskedastic models". In: *Journal of Economic Dynamics and Control* 18.5, pp. 931–955. DOI: 10.1016/0165-1889(94)90039-6.