

Alwin Peters

Pelikaanhof 72B, 2312 EC, Leiden
alwinpeters@hotmail.com
s1047558

Tijdreeksen van koffie-futures

Bachelorscriptie, 5 november 2013

Scriptiebegeleider: prof.dr. A.W. van der Vaart



Mathematisch Instituut, Universiteit Leiden

Inhoudsopgave

Inleiding	i
0.1 Samenvatting	i
0.1.1 Details	iii
1 ARMA-modellen	1
1.1 Stochastisch proces	1
1.1.1 Stationair	1
1.1.2 Ergodiciteit	2
1.1.3 Wold's decomposition	3
1.2 ARMA processen	4
1.2.1 AR(1)	4
1.2.2 MA(1)	5
1.2.3 AR(p)	6
PACF	7
1.2.4 MA(q)	8
1.2.5 ARMA(p,q)	9
1.2.6 Data , deel 1	9
1.3 Criteria	11
1.3.1 Ljung and Box	11
1.3.2 Akaike's Information Criteria	11
1.3.3 AICc	12
1.3.4 Bayesian Information Criteria / Schwarz Criteria	12
1.3.5 Vergelijking AIC, AICc en BIC	12
1.3.6 Data, deel 2	13
1.4 ARIMA	14

<i>INHOUDSOPGAVE</i>	3
2 Unit roots	15
2.0.1 Dickey-Fuller test	15
2.0.2 Data, deel 3	19
2.0.3 Data, deel 4	22
2.0.4 Eerste conclusie	23
3 GARCH-modellen	24
3.0.5 Martingaal	24
3.0.6 ARCH(1)	24
Kurtosis van ARCH(1)	25
3.0.7 Data, deel 5	26
Variantie van innovaties van ARCH(1)	31
3.0.8 ARCH(q)	32
3.0.9 GARCH(p,q)	32
3.0.10 Modificaties van GARCH	32
sGARCH	33
iGARCH	33
apARCH	33
eGARCH	33
GJRGARCH	34
TGARCH	34
AVGARCH	34
NGARCH of NAGARCH	34
3.0.11 Data, deel 6	34
3.0.12 Data, deel 7	35
3.0.13 Data, deel 9	35
3.1 Mogelijkheden	38
3.1.1 VEC specificatie van GARCH(p,q)	39
3.1.2 BEKK	40
3.1.3 CCC	40
3.1.4 Verklarend gedrag en theorie rond sprongen	40
3.1.5 Eindconclusie	43
Appendix	45
.1 Scripts	45
.1.1 Script 1	45
.1.2 Script 2	46

INHOUDSOPGAVE

4

.1.3	Script 3	46
.1.4	Script 4	46
.1.5	Script 5	47
.1.6	Script 6	47
.1.7	Script 7	48
.1.8	Script 8	49
.1.9	Script 9	49
.1.10	Script 10	50
.1.11	Script 11	51
.1.12	Script 12	51
.1.13	Script 13	51
.1.14	Script 14	57
.1.15	Script 15	59
.1.16	Script 16	59
.1.17	Script 17	67
.2	Figuren	68

Inleiding

In deze scriptie zal worden ingegaan op een probleem van NEDCoffee, een Nederlands bedrijf, dat koffie distribueert. Dit bedrijf is beursgenoteerd, zowel in New York als in Londen. Het bedrijf heeft contracten van een tijdsduur van 2, 3 en 4 maanden. De koers van dat moment, zeg P_0 , zal dan de verkoopprijs zijn (met een kleine verhoging om het risico te verlagen), terwijl de prijs op het moment van levering van het contract, over het algemeen anders zal zijn, zeg P_1 .

Momenteel is het model van NEDCoffee niet toereikend om het verlies te verkleinen, dan wel de winst te vergroten. In deze scriptie zal dieper worden ingegaan op de wiskunde achter deze koersen en zal worden geprobeerd om een model te vinden dat bij deze koers hoort. Het uiteindelijke model geeft een goed passend model bij de data die het bedrijf heeft vrijgegeven voor dit onderzoek.

Daarnaast was het probleem dat de data in de door NEDCoffee veronderstelde modellen normaal verdeeld zouden zijn, terwijl dat dit niet het geval is. Hier zal ook nog kort naar worden gekeken.

0.1 Samenvatting

In deze scriptie wordt ingegaan op de wiskunde die achter de tijdreeksen zit. Allereerst zal er een korte introductie zijn over een aantal belangrijke aannames, zoals ergodiciteit en stationair gedrag van een stochastisch proces. Daarna zullen verschillende karakteristieken van zo'n stochastisch proces worden uitgelegd met behulp van de wiskunde die hier achter steekt. Hierbij wordt kort ingegaan op de variantie en de verwachtingswaarde en daarnaast zal er aandacht worden besteed aan de AutoCorrelation Function en de Partial AutoCorrelation Function.

Met behulp hiervan wordt er gekeken naar de AutoRegressive Moving Average-modellen en daarna wordt er gekeken naar de verschillende criteria voor dit soort modellen. Met behulp van die criteria kan het model worden getoetst, met als nulhypothese dat de 'afwijking' als white noise verdeeld is. Dan kan er worden geconcludeerd dat dit niet het geval is en daarom zal er gebruik worden gemaakt van een ander type model, namelijk het AutoRegressive Integrated Moving Average-model.

Voor dit model kunnen weer dezelfde toetsen en criteria worden gebruikt en ook hier valt op dat de afwijking niet als white noise is verdeeld. Om met zekerheid te kunnen stellen dat er inderdaad een "Integrated-model moet worden gebruikt, zullen er verschillende testen zoals de (augmented) Dickey-Fuller test en de Phillips-Perron test worden geïntroduceerd. Deze wijzen uit dat er een "Integrated-model nodig is.

Daarna zal er een model worden geïntroduceerd om de eerder genoemde 'afwijking' te simuleren, deze modellen heten (Generalised) AutoRegressive Conditional Heteroskedasticity-modellen. Hier zijn dan ook verschillende variaties op te vinden en deze zullen ook kort worden geïntroduceerd. Tussentijds zal in alle hoofdstukken de data van NEDCoffee worden gebruikt om verschillende modellen te beschrijven, allereerst de ARIMA-modellen in het eerste hoofdstuk, de verschillende testen hierop zullen dan worden uitgevoerd in alle hoofdstukken en dan zal de data met behulp van de GARCH modellen en variaties hierop worden gebruikt om tot een passend model te komen. Dit gebeurt in het derde hoofdstuk. Tevens zal er in dat hoofdstuk nog wat over de vorm van de tijdreeksen voorbij komen.

Ten slotte worden de beste modellen gebruikt om de deelproblemen op te lossen en uit te zoeken welke modellen er bij de gehele data passen. Dit zal in hoofdstuk 3 terug te lezen zijn. Daar zal ook nog een aantal mogelijkheden voor passende modellen worden genoemd, die verder als suggestie staan gegeven. Tevens introduceer ik daar ook nog een apart vermoeden.

De gebruikte scripts staan in de appendix, net als de figuren.

0.1.1 Details

Hieronder kunt u de gebruikte afkortingen en dergelijke vinden:

- ACF: AutoCorrelation Function
- PACF: Partial AutoCorrelation Function
- AR: AutoRegressive
- MA: Moving Average
- AR(I)MA: AutoRegressive (Integrated) Moving Average
- SACF: Sample AutoCorrelation Function
- SPACF: Sample Partial AutoCorrelation Function
- AIC: Akaike's Information Criteria
- AICc: Akaike's Information Criteria with a correction
- BIC: Bayesian Information Criteria / Schwarz Criterion
- OLS : Ordinary Least Square
- WN : White Noise
- SWN : Strikt White Noise
- NID : Normally Independent Distributed
- (G)ARCH: (Generalised) AutoRegressive Conditional Heteroskedasticity
- VEC : Vectorisatie
- VECH : Half-vectorisatie
- BEKK : Acronym van Baba, Engle, Kraft en Kroner
- CCC : Constant conditional correlation

- Dataset : $\{x_t\}_1^T = (x_1, \dots, x_T)$
- Stochastisch proces : $\{X_t\}_1^T = (X_1, \dots, X_T)$
- Verwachtingswaarde : $E(X_t)$ of μ
- Variantie : $V(X_t)$ of σ_X^2
- Covariantie : $Cov(X_i, X_j)$
- Autovariantie : γ_k
- Autocorrelatie : ρ_k
- Innovaties : a_t
- Backshift operator : B
- Wiener Proces / Brownian Motion : $W(r)$
- Chi-square verdeling (met q vrijheidsgraden) : χ_q^2
- Kurtosis : κ

Hoofdstuk 1

ARMA-modellen

In dit hoofdstuk wordt gekeken naar wat simpelere modellen, waarbij de basisprincipes voor het bekijken van de koers makkelijker uiteen te zetten zijn. Hierbij zullen met name de AR, MA, ARMA en ARIMA processen en de bijbehorende ACF's en PACF's worden bekeken.

Daarna komen de verschillende testen, zoals AIC, Ljung-Box en BIC testen, aan bod.

1.1 Stochastisch proces

Kijk naar een serie van data (x_1, x_2, \dots, x_T) , ook wel $\{X_t\}_1^T$. Het gehele proces heeft T verwachtingen, namelijk $E(X_1), E(X_2), \dots, E(X_T)$ en T varianties, namelijk $V(X_1), V(X_2), \dots, V(X_T)$. Verder zijn er nog $\frac{T(T-1)}{2}$ covarianties, namelijk $Cov(X_i, X_j), i < j$.

1.1.1 Stationair

Dit houdt in dat een proces zich in een bepaalde staat van een evenwicht bevindt. Als een stochastisch proces strikt stationair is, dan zijn alle eigenschappen onveranderd door de tijd. Oftewel, de gezamenlijke kansverdeling van $X_{t_1}, X_{t_2}, \dots, X_{t_m}$ is dezelfde als de gezamenlijke kansverdeling van $X_{t_1+k}, X_{t_2+k}, \dots, X_{t_m+k}$ met k een bepaalde tijdstap (die aan randvoorwaarden en verder eigenschappen van het proces voldoet). Hieruit volgt dat als $m = 1$, dat dan zolang $E|X_t|^2 < \infty$, dat dan de verwachtingswaarde en

variantie van X_t constant blijven, oftewel

$$E(X_1) = E(X_2) = \dots = E(X_T) = E(X_t) = \mu$$

$$V(X_1) = V(X_2) = \dots = V(X_T) = V(X_t) = \sigma_x^2$$

Uit $m = 2$ volgt dan dat alle bivariate verdeling onafhankelijk van t zijn, dus alle covarianties zijn functies van die k , oftewel

$$\text{Cov}(X_1, X_{1+k}) = \text{Cov}(X_2, X_{2+k}) = \dots = \text{Cov}(X_{T-k}, X_T) = \text{Cov}(X_t, X_{t+k})$$

Nu kan op basis hiervan de autocovarianties als volgt worden gedefiniëerd:

$$\gamma_k = \text{Cov}(X_t, X_{t-k}) = E[(X_t - \mu)(X_{t-k} - \mu)]$$

en ook de autocorrelaties als volgt definiëren:

$$\rho_k = \frac{\text{Cov}(X_t, X_{t-k})}{[V(X_t) \cdot V(X_{t-k})]^{1/2}} = \frac{\gamma_k}{\gamma_0}$$

1.1.2 Ergodiciteit

Zij X_t een stationair proces. X_t heet dan ergodisch als

$$\frac{1}{n} \sum_{t=1}^n f(X_t) \rightarrow_p E(f(X_1)) \text{ voor alle } f \text{ waarvoor } E(f(X_1)) \text{ bestaat.}$$

Hierbij betekent \rightarrow_p dat dit met een bijna zekere kans convergeert.

Opmerking 1.1 (Variantie en ACF).

$$V(X_t) = V(X_{t-k})$$

$$\gamma_0 = E[(X_t - \mu)^2] = V(X_t)$$

$$\gamma_k = \text{Cov}(X_t, X_{t-k}) = \text{Cov}(X_{t-k}, X_t) = \text{Cov}(X_t, X_{t+k}) = \gamma_{-k}$$

Dus dan volgt dat $\rho_k = \rho_{-k}$ en dus hoeft er alleen te worden gekeken naar de positieve helft van de ACF.

Opmerking 1.2 (Karakteristieken). Voor het beschrijven en modelleren van observaties, kan worden gekeken naar karakteristieken, als:

$$\mu = E(X_t)$$

$$\sigma_x^2 = \gamma_0 = V(X_t)$$

ACF

1.1.3 Wold's decomposition

Elke zwak stationaire, pure niet-deterministische, stochastisch proces ($X_t - \mu$) is te schrijven als een lineaire combinatie van een rij van ongecorrleerde random innovaties.

Definitie 1.1 (Lineaire filter Representatie).

$$X_t - \mu = a_t + \psi_1 a_{t-1} + \psi_2 a_{t-2} + \dots = \sum_{j=0}^{\infty} \psi_j a_{t-j}, \psi_0 = 1$$

De verzameling $\{a_t : t = 0, \pm 1, \pm 2, \dots\}$ is een rij van ongecorrleerde random variabelen, ook wel innovaties, van een verdeling met :

$$\text{Eigenschap 1: } E(a_t) = 0 \text{ (i)}$$

$$\text{Eigenschap 2: } V(a_t) = E(a_t^2) - (E(a_t))^2 = E(a_t^2) = \sigma^2 < \infty \text{ (ii)}$$

$$\begin{aligned} \text{Eigenschap 3: } \text{Cov}(a_t, a_{t-k}) &= E\left((a_t - E(a_t))(a_{t-k} - E(a_{t-k}))\right) \\ &= E(a_t a_{t-k}) = 0, \text{ voor alle } k \neq 0 \text{ (iii)} \end{aligned}$$

Notatie 1.1 (White Noise proces). Dit wordt genoteerd als $a_t \sim \text{WN}(0, \sigma^2)$.

Het model geeft autocorrelatie in X_t :

$$E(X_t) = E(\mu) + E(\psi_0 a_t) + E(\psi_1 a_{t-1}) + \dots = \mu + \psi_0 E(a_t) + \psi_1 E(a_{t-1}) + \dots =_i \mu$$

$$\gamma_0 = V(X_t) = E((X_t - \mu)^2) = E((a_t + \psi_1 a_{t-1} + \psi_2 a_{t-2} + \dots)^2)$$

$$=_{iii} E(a_t^2) + \psi_1^2 E(a_{t-1}^2) + \psi_2^2 E(a_{t-2}^2) + \dots =_{ii} 1 \cdot \sigma^2 + \psi_1^2 \cdot \sigma^2 + \psi_2^2 \cdot \sigma^2 + \dots = \sigma^2 \sum_{j=0}^{\infty} \psi_j^2$$

$$\gamma_k = E\left((X_t - E(X_t))(X_{t-k} - E(X_{t-k}))\right)$$

$$=_i E\left((a_t + \psi_1 a_{t-1} + \psi_2 a_{t-2} + \dots)(a_{t-k} + \psi_1 a_{t-k-1} + \psi_2 a_{t-k-2} + \dots)\right)$$

$$=_{iii} \psi_k E(a_{t-k}^2) + \psi_1 \psi_{1+k} E(a_{t-k-1}^2) + \psi_2 \psi_{2+k} E(a_{t-k-2}^2) + \dots$$

$$=_{ii} \sigma^2 (\psi_0 \psi_k + \psi_1 \psi_{1+k} + \psi_2 \psi_{2+k} + \dots) = \sigma^2 \sum_{j=0}^{\infty} \psi_j \psi_{j+k}$$

$$\rho_k = \frac{\gamma_k}{\gamma_0} = \frac{\sum_{j=0}^{\infty} \psi_j \psi_{j+k}}{\sum_{j=0}^{\infty} \psi_j^2}$$

Als er een oneindig aantal ψ -gewichten zijn, dan moet worden aangenomen dat de gewichten absoluut sommeerbaar zijn, dus dat $\sum_{j=0}^{\infty} |\psi_j| < \infty$. Dan convergeert de lineaire filter representatie.

Dan volgt dat X_t stationair is en dat $\gamma_0 = V(X_t)$ eindig is.

Met behulp van de volgende stelling is te valideren dat de manier van schrijven van de lineaire filter representatie als in definitie 1.1 juist is.

Stelling 1.1 (Combinatie). *Als $\sum_{j=0}^{\infty} |\psi_j| < \infty$ en $\sup_t E(|a_t|) < \infty$, dan convergeert $\sum_{j=0}^{\infty} \psi_j a_{t-j}$ bijna zeker. Dan volgt dat dit ook geldt in "L₂", $E(|\sum_{j>k} \psi_j a_{t-j}|)^2 \rightarrow 0$ als $k \rightarrow \infty$.*

[9]

1.2 ARMA processen

1.2.1 AR(1)

Beschouw de lineaire filter representatie als in definitie 1.1. Neem $\mu = 0$, dit geeft geen verlies van generaliteit (hier wordt later op teruggekomen), en kies $\psi_j = \phi^j$. Dan wordt de lineaire filter representatie:

$$X_t = a_t + \phi a_{t-1} + \phi^2 a_{t-2} + \dots = a_t + \phi(a_{t-1} + \phi a_{t-2} + \dots) = a_t + \phi X_{t-1}$$

$$\text{Oftewel, } X_t - \phi X_{t-1} = a_t$$

Notatie 1.2 (AR(1) proces). *Dit wordt genoteerd als AR(1).*

Definitie 1.2 (Backshift operator). *B is de backshift operator met de volgende eigenschap:*

$$BX_t \equiv X_{t-1};$$

$$B^m X_t \equiv X_{t-m}$$

Dus AR(1) is te schrijven als $(1 - \phi B)X_t = a_t$.

$$\text{Dus } X_t = (1 - \phi B)^{-1} a_t = (1 + \phi B + \phi^2 B^2 + \dots) a_t = a_t \sum_{j=0}^{\infty} \phi^j B^j$$

De voorwaarde hiervoor is dat $|\phi| < 1$. Dat volgt uit het volgende:

$$(1 - \phi B)(1 + \phi B + \phi^2 B^2 + \dots + \phi^m B^m) = 1 - \phi^{m+1} B^{m+1}$$

Omdat hier wordt gekeken naar een oneindig aantal gewichten, moet dus gelden dat

$$\lim_{m \rightarrow \infty} \phi^{m+1} \rightarrow 0, \text{ dus } |\phi| < 1$$

Dit heet ook wel de stationaire conditie.

Op de volgende manier wordt geprobeerd om de ACF van het AR(1) model vinden. Daarvoor moet er worden vermenigvuldigd met $X_{t-k}, k > 0$.

$$X_{t-k}(X_t - \phi X_{t-1}) = X_{t-k}a_t$$

Hiervan moeten dan de verwachtingen worden bepaald, dat geeft dan

$$\gamma_k - \phi\gamma_{k-1} = E(X_{t-k}a_t)$$

$$a_t X_{t-k} = a_t(a_{t-k} + \phi a_{t-k-1} + \phi^2 a_{t-k-2} + \dots) = \sum_{i=0}^{\infty} a_t a_{t-k-i} \phi^i$$

En eigenschap 3, te vinden op pagina 3, geeft dat dan $E(X_{t-k}a_t) = 0$. Dus dan volgt dat

$$\gamma_k = \phi\gamma_{k-1}, \text{ voor alle } k > 0$$

Dus $\gamma_k = \phi^k \gamma_0$

Dus de ACF is : $\rho_k = \frac{\gamma_k}{\gamma_0} = \phi^k$

1.2.2 MA(1)

Beschouw de lineaire filter representatie, zie definitie 1.1. Neem $\mu = 0$, dit geeft geen verlies van generaliteit (hier wordt later nog op teruggekomen), en kies $\psi_1 = -\theta$ en $\psi_j = 0$ voor $j \geq 2$. Dan wordt de lineaire filter representatie:

$$X_t = a_t - \theta a_{t-1} = a_t(1 - \theta B)$$

Notatie 1.3 (MA(1) proces). *Dit wordt genoteerd als MA(1).*

Gebruik makend van de eigenschappen 1, 2 en 3, te vinden onder definitie 1.1, volgt dat:

$$\mu = E(X_t) = 0$$

$$\gamma_0 = V(X_t) = E\left((X_t - E(X_t))^2\right) = E(X_t^2) = E((a_t - \theta a_{t-1})(a_t - \theta a_{t-1}))$$

$$\begin{aligned}
&= E(a_t^2) + \theta^2 E(a_{t-1}^2) = \sigma^2(1 + \theta^2) \\
\gamma_1 &= E\left((X_t - E(X_t))(X_t - E(X_{t-1}))\right) = E(X_t X_{t-1}) \\
&= E\left((a_t - \theta a_{t-1})(a_{t-1} - \theta a_{t-2})\right) = -\theta E(a_{t-1}^2) = -\theta \sigma^2 \\
\gamma_k &= E(X_t X_{t-k}) = E\left((a_t - \theta a_{t-1})(a_{t-k} - \theta a_{t-k-1})\right) = 0
\end{aligned}$$

Dus de ACF is

$$\rho_1 = \frac{\gamma_1}{\gamma_0} = -\frac{\theta}{1 + \theta^2}$$

en $\rho_k = 0$, voor $k > 1$.

Omdat er een eindig aantal ψ -gewichten zijn in het MA-model, is het MA-model stationair. Dit kan ook wel worden gerepresenteerd als een $AR(\infty)$, als $|\theta| < 1$. Omdat er een eigenschap is die vertelt dat het systeem invertibeerbaar is, het ook wel te schrijven als:

$$X_t = \pi_1 X_{t-1} + \pi_2 X_{t-2} + \dots + at = \sum_{j=1}^{\infty} \pi_j X_{t-j} + a_t$$

waarbij de π -gewichten convergeren, oftewel, er geldt dat $\sum_{j=1}^{\infty} \pi_j < \infty$.

Dus het MA(1)-model is als volgt te beschouwen:

$$X_t = a_t - \theta a_{t-1} = (1 - \theta B)a_t$$

$$a_t = (1 - \theta B)^{-1} X_t = (1 + \theta B + \theta B^2 + \dots) X_t = a_t$$

Dan volgt dat $\pi_j = -\theta^j$ convergeert als $\theta < 1$, dus het systeem is invertibeerbaar.

1.2.3 AR(p)

Een AR(p)-model is te schrijven als

$$X_t - \phi_1 X_{t-1} - \dots - \phi_p X_{t-p} = (1 - \phi_1 B - \dots - \phi_p B^p) X_t = \phi(B) X_t = a_t$$

Notatie 1.4 (AR(p) proces). Dit wordt genoteerd als $AR(p)$.

Opmerking 1.3. Dit is geen unieke manier van X_t schrijven.

Zij nu X_1, X_2, \dots, X_p gegeven.

Dan is $\phi(z) = 1 - \phi_1 z - \phi_2 z^2 - \dots - \phi_p z^p$.

Dan is er een stationaire oplossing van het AR(p) model met randvoorwaarden X_1, \dots, X_p dan en slechts dan als $\phi(z) \neq 0$ voor $z \in \mathbf{C}, |z| = 1$.

De oplossing is dan uniek en causaal dan en slechts dan als $\phi(z) \neq 0$ voor $z \in \mathbf{C}, |z| \leq 1$, wat equivalent is met zeggen dat alle nulpunten van $\phi(z)$ buiten de eenheidskring liggen.

Dit geeft dan dat $\frac{1}{\phi(z)} = \sum_{j=0}^{\infty} \psi_j z^j$ absoluut convergeert en in het bijzonder voor $|z| = 1$, wat dan weer equivalent is met zeggen dat $\sum_{j=0}^{\infty} |\psi_j| |z|^j = \sum_{j=0}^{\infty} |\psi_j| < \infty$.

Opmerking 1.4 (Causaal). Causaal betekent dat $X_t = \sum_{j \geq 0} \psi_j a_{t-j}$ voor zekere ψ_j

[9]

PACF

Bekijk nu het AR(p)-model als hiervoor beschreven en vermenigvuldig dit met $X_{t-i}, 1 \leq i \leq p$ en neem dan verwachtingen. Dat geeft dan voor $1 \leq i \leq p$:

$$E(X_t X_{t-i}) = \sum_{j=1}^p \phi_j E(X_{t-j} X_{t-i}) + E(X_{t-i} a_t)$$

$E(X_{t-i} a_t) = 0$, want a_t is ongecorrleerd met voorgaande waardes.

$$E(X_t X_{t-i}) = \gamma_i \text{ en } E(X_{t-j} X_{t-i}) = \gamma_{j-i}$$

$$\text{Dus dan volgt dat } \gamma_i = \sum_{j=1}^p \phi_j \gamma_{j-i}$$

$$\text{Delen door } \gamma_0 \text{ geeft dan dat } \rho_i = \sum_{j=1}^p \phi_j \rho_{j-i}$$

Gebruik makend van opmerking 1.1, geeft dan dat $\rho_{-i} = \rho_i, 1 \leq i \leq p$ en verder geldt dat $\rho_0 = 1$. Dus dan volgt er het volgende stelsel

$$\begin{pmatrix} \rho_1 \\ \rho_2 \\ | \\ | \\ | \\ \rho_{p-1} \\ \rho_p \end{pmatrix} = \begin{pmatrix} \phi_1 & + & \phi_2 \rho_1 & + & \dots & + & \phi_{p-1} \rho_{p-2} & + & \phi_p \rho_{p-1} \\ \phi_1 \rho_1 & + & \phi_2 & + & \dots & + & \phi_{p-1} \rho_{p-3} & + & \phi_p \rho_{p-2} \\ | & & | & & & & | & & | \\ | & & | & & & & | & & | \\ | & & | & & & & | & & | \\ \phi_1 \rho_{p-2} & + & \phi_2 \rho_{p-3} & + & \dots & + & \phi_{p-1} & + & \phi_p \rho_1 \\ \phi_1 \rho_{p-1} & + & \phi_2 \rho_{p-2} & + & \dots & + & \phi_{p-1} \rho_1 & + & \phi_p \end{pmatrix}$$

Zij nu $p = k$ en zij $\phi_i = \phi_{ii}$, $1 \leq i \leq p$. Dan volgt dat $P_{(k)} = P_k \phi_k$, waarbij $P_{(k)} = (\rho_1, \rho_2, \dots, \rho_{k-1}, \rho_k)^T$, $\phi_k = (\phi_{11}, \phi_{22}, \dots, \phi_{k-1, k-1}, \phi_{kk})^T$ en $P_k =$

$$\begin{pmatrix} 1 & \rho_1 & \dots & \rho_{k-2} & \rho_{k-1} \\ \rho_1 & 1 & \dots & \rho_{k-3} & \rho_{k-2} \\ | & | & & | & | \\ | & | & & | & | \\ | & | & & | & | \\ \rho_{k-2} & \rho_{k-3} & \dots & 1 & \rho_1 \\ \rho_{k-1} & \rho_{k-2} & \dots & \rho_1 & 1 \end{pmatrix}$$

Dan geeft Cramer's Rule dat $\phi_{kk} = \frac{|P_k^*|}{|P_k|}$, waarbij P_k^* uit P_k ontstaat door de laatste kolom te vervangen door $P_{(k)}$.

1.2.4 MA(q)

Een MA(q)-model is te schrijven als

$$X_t = a_t - \theta_1 a_{t-1} - \dots - \theta_q a_{t-q} = (1 - \theta_1 B - \dots - \theta_q B^q) a_t = \theta(B) a_t$$

Notatie 1.5 (MA(q) proces). *Dit wordt genoteerd als MA(q).*

Om de ACF hiervan te bepalen, kan er worden vermenigvuldigd met X_{t-k} en daarna verwachtingen genomen geworen. Dat geeft dan:

$$\begin{aligned} E(X_t X_{t-k}) &= E(a_t - \theta_1 a_{t-1} - \dots - \theta_k a_{t-k} - \dots - \theta_q a_{t-q})(a_{t-k} - \\ &\quad \theta_1 a_{t-k-1} - \dots - \theta_{q-k} a_{t-q} \dots - \theta_q a_{t-k-q}) \\ \gamma_k &= -\theta_k E(a_{t-k}^2) + \theta_1 \theta_{k+1} E(a_{t-k-1}^2) + \dots + \theta_{q-k} \theta_q E(a_{t-q}^2) \\ \gamma_k &= \sigma^2(-\theta_k + \theta_1 \theta_{k+1} + \dots + \theta_{q-k} \theta_q), \text{ voor } 1 \leq k \leq q \\ \gamma_0 &= \sigma^2(1 + \theta_1^2 + \dots + \theta_q^2) \\ \rho_k &= \frac{-\theta_k + \theta_1 \theta_{k+1} + \dots + \theta_{q-k} \theta_q}{1 + \theta_1^2 + \dots + \theta_q^2}, \text{ voor } 1 \leq k \leq q \\ \rho_k &= 0 \text{ voor } k > q \end{aligned}$$

Dit model is dan inverteerbaar, dus ook te schrijven als

$$(1 - \theta_1 B - \dots - \theta_q B^q) = (1 - h_1 B)(1 - h_2 B) \dots (1 - h_q B) = 0$$

met $|h_i| < 1$, $1 \leq i \leq q$.

1.2.5 ARMA(p,q)

Een ARMA(p,q)-model is te schrijven als

$$\begin{aligned} X_t - \phi_1 X_{t-1} - \dots - \phi_p X_{t-p} &= a_t - \theta_1 a_{t-1} - \dots - \theta_q a_{t-q} \\ (1 - \phi_1 B - \dots - \phi_p B^p) X_t &= (1 - \theta_1 B - \dots - \theta_q B^q) a_t \\ \phi(B) X_t &= \theta(B) a_t \end{aligned}$$

Notatie 1.6 (ARMA(p,q) proces). *Dit wordt genoteerd als MA(q).*

Opmerking 1.5 (Voorwaardes). *Dit proces heeft dezelfde stationaire en inverseerbare voorwaardes als AR(p) en MA(q).*

Opmerking 1.6 (ACF en PACF). *ACF volgt AR(p) patroon na q - p waardes. PACF volgt MA(q) patroon na p - q waardes.*

Opmerking 1.7 (μ). *Als $\mu \neq 0$, volgt dat*

$$\begin{aligned} \phi(B)(X_t - \mu) &= \theta(B)a_t \\ \phi(B)\mu &= (1 - \phi_1 B - \dots - \phi_p B^p)\mu = (1 - \phi_1 - \dots - \phi_p)\mu = \phi(1)\mu \\ \text{Dus } \phi(B)X_t &= \theta_0 + \theta(B)a_t, \theta_0 = \phi(1)\mu \end{aligned}$$

Opmerking 1.8. *Als $a_t \sim WN(0, \sigma^2)$ ook nog onafhankelijk verdeeld zijn, dan is a_t strikt white noise.*

Notatie 1.7 (Strikt White Noise). $a_t \sim SWN(0, \sigma^2)$

[9]

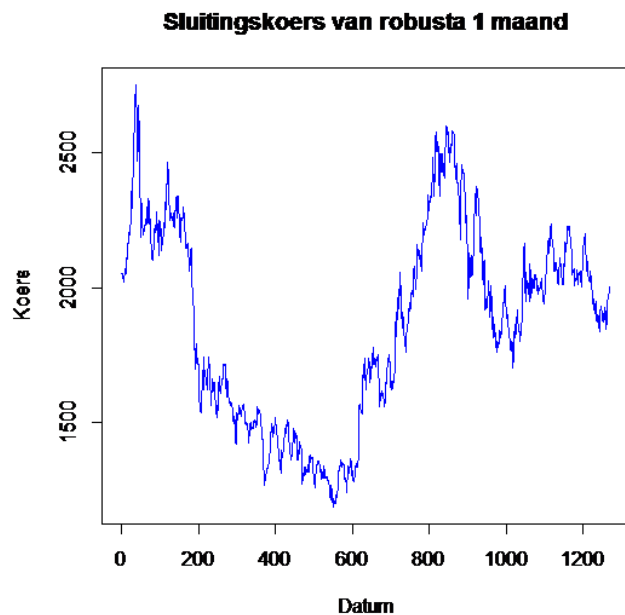
1.2.6 Data , deel 1

Kijk nu naar de 2^e kolom van de robusta.txt file.

Deze tijdreeks ziet eruit als de plot te vinden op de volgende pagina, figuur 1.1.

Neem nu aan dat deze set ergodisch (1.1.2) en stationair (1.1.1) is.

Opmerking 1.9 (Stationair). *De set lijkt momenteel niet stationair te zijn.*



Figuur 1.1: Plot van de robusta set, 2e kolom

Schat dan van deze set de verwachtingswaarde en variantie, door van de realisatie $\{x_t\}_1^T$ het volgende te nemen:

$$\bar{x} = \frac{1}{T} \sum_{t=1}^T x_t$$

$$s^2 = \frac{1}{T} \sum_{t=1}^T (x_t - \bar{x})^2$$

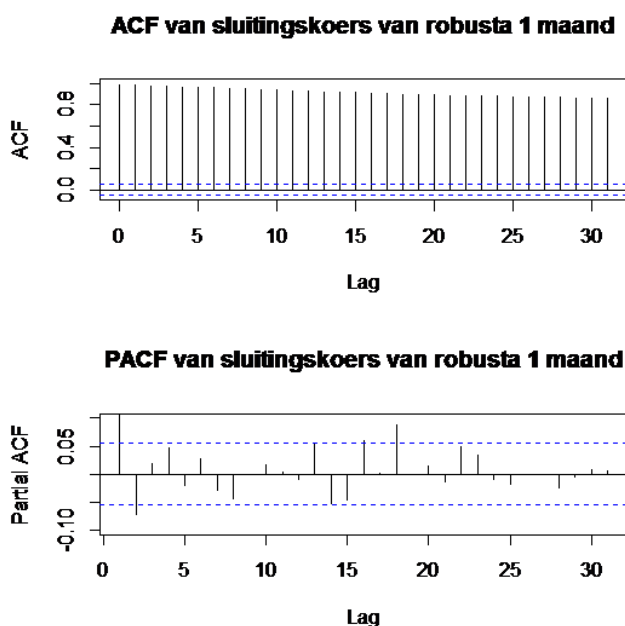
Zo is de ACF ook te schatten, door

$$r_k = \frac{\sum_{t=k+1}^T (x_t - \bar{x})(x_{t-k} - \bar{x})}{Ts^2}, k \leq 1$$

De verzameling van deze r_k 's is dan de SACF.

De SPACF is dan telkens de schatter van de laatste coëfficiënt in het AR(p) model, $\hat{\phi}_{kk}$.

Om dit alles voor deze set te bepalen, kan het script worden gebruikt dat te vinden is in de appendix, pagina 45, .1.1. waarbij de plot van de ACF en PACF als volgt is:



Figuur 1.2: Plot van de acf en pacf van de robusta set, 2e kolom

Opmerking 1.10 (PACF). *Op de plot van de PACF is aan te merken dat er een waarde voor $k = 0$ is te vinden, maar die is natuurlijk 1, vandaar dat die waarde in de plot wel terug te zien is, maar niet in zijn geheel, deze is namelijk niet interessant.*

1.3 Criteria

1.3.1 Ljung and Box

De nulhypothese is dat de returns verdeeld zijn als white noise. Om dit te bepalen, kan de volgende test worden gebruikt:

$$Q(k) = T(T+2) \sum_{i=1}^k (T-i)^{-1} r_i^2 \sim^a \chi_k^2$$

1.3.2 Akaike's Information Criteria

$$AIC(p, q) = 2k - 2\ln(L)$$

Waarbij k het aantal parameters in het model is en L de likelihood.

Notatie 1.8 (AIC). *Dit wordt genoteerd als AIC.*

Deze waarde moet dan over alle mogelijke modellen voor $0 \leq p \leq p_{max}$ en $0 \leq q \leq q_{max}$ worden geminimaliseerd.

1.3.3 AICc

Dit is een gecorrigeerde AIC voor eindige groottes van datasets:

$$AICc = AIC + \frac{2k(k+1)}{n-k-1} = 2k - 2\ln(L) + \frac{2k(k+1)}{n-k-1}$$

Waarbij k het aantal parameters in het model is, L de likelihood en n de grootte van de dataset.

Notatie 1.9 (AICc). *Dit wordt genoteerd als AICc.*

Deze waarde moet dan over alle mogelijke modellen voor $0 \leq p \leq p_{max}$ en $0 \leq q \leq q_{max}$ worden geminimaliseerd.

1.3.4 Bayesian Information Criteria / Schwarz Criteria

$$BIC(p, q) = k \cdot \ln(n) - 2\ln(L)$$

Waarbij k het aantal parameters in het model is, L de likelihood en n de grootte van de dataset.

Notatie 1.10 (BIC). *Dit wordt genoteerd als BIC.*

Deze waarde moet dan over alle mogelijke modellen voor $0 \leq p \leq p_{max}$ en $0 \leq q \leq q_{max}$ worden geminimaliseerd. [9]

1.3.5 Vergelijking AIC, AICc en BIC

Als eerste is het handig op te merken dat het verschil tussen de AIC en de AICc heel klein is. Zo is ook te vinden dat als $n \rightarrow \infty$ dat dan $AIC \rightarrow AICc$. Zo is ook direct te zien dat de AIC en AICc verschillende modellen zullen kiezen voor kleine n . In deze scriptie zijn ze allebei weergegeven en geven de AIC en AICc telkens dezelfde modellen.

Omdat de $AIC(p, q) = 2k - 2\ln(L)$ en $BIC(p, q) = k \cdot \ln(n) - 2\ln(L)$ zien we dat het verschil zit in de factoren $2k$ en $k \cdot \ln(n)$. Dit is wiskundig een erg groot verschil. Omdat de waardes worden geminimaliseerd en de

waarde $\ln(n)$ erg groot kan zijn, zal de BIC de eigenschap hebben om een model met weinig parameters te kiezen, terwijl dat dit voor de AIC minder het geval zal zijn.

Kwalitatief gezien is het grootste verschil dat de AIC een model geeft dat als volgt is te beschouwen:

Zij X_1, X_2, \dots, X_n waargenomen dat een model geeft met als parameters $(p, \hat{\theta}_p)$. Het minimaliseren geeft dan een model met \hat{p} en dan kan X_{n+1} worden voorspeld met het model $(\hat{\theta}_{\hat{p}}, \hat{p})$, wat dan een likelihood geeft die als volgt is te noteren: $\prod(X_1, \dots, X_n, \hat{\theta}_{\hat{p}}, \hat{p})$.

De AIC geeft dan de beste voorspelling die $\prod(X_1, \dots, X_n, \hat{\theta}_{\hat{p}}, \hat{p})$ minimaliseert.

Dit kijkt in feite dan naar $E(X_{n+1} - \prod(X_1, \dots, X_n, \hat{\theta}_{\hat{p}}, \hat{p}))^2$.

De BIC daarentegen heeft de eigenschap om het model voor de voorgaande het beste te schatten.

Vandaar dat er in deze scriptie geselecteerd zal worden met de AIC (of AICc). Het doel dat hier wordt aangenomen is daarbij dan ook het zoeken naar een model waarmee de toekomstige waardes kunnen worden geschat.

1.3.6 Data, deel 2

De Ljung-Box test kan worden gebruikt om te testen of er mag worden aangenomen dat de returns van de robustaset verdeeld zijn als white noise.

Dat kan worden gedaan met het script te vinden op pagina 46, .1.2.

Opmerking 1.11 (Box-Ljung test). *Daarvoor kan ook een functie uit R worden aangeroepen (nu ter controle), deze is te vinden op pagina 46, .1.3. P is dan hetzelfde als Q1.*

Zo worden er allemaal p-waardes verkregen die bijna 0 zijn, wat dus alle redenen geeft om te twijfelen aan de nullhypothese dat de returns als white noise zijn verdeeld. Later zal veelvuldig gebruik worden gemaakt van de AIC, de AICc en de BIC.

1.4 ARIMA

Zij d een natuurlijk getal.

Definitie 1.3 (Difference function). $\Delta^d X_t = (1 - B)^d X_t$

Dan is

$$\phi(B)\Delta^d X_t = \theta_0 + \theta(B)a_t$$

Notatie 1.11 (ARIMA). Ook wel $ARIMA(p,d,q)$.

[9]

Opmerking 1.12. d kan ook een niet-natuurlijk getal zijn, zo'n type model heet een $ARFIMA(p,d,q)$ -model. Dit lijkt hier alleen niet zinvol te zijn, dus dit wordt verder buiten beschouwing gelaten.

Hoofdstuk 2

Unit roots

In dit hoofdstuk zal kort ingegaan worden op de unit-roots en de toepassingen hiervan zullen ook worden beschreven.

Met behulp van een drietal testen zal er worden gezocht naar de unit roots in het model en zo zal dan worden bepaald wat de orde is van de d uit het ARIMA(p, d, q)-model. Die drie testen zijn de Dickey-Fuller test (DF), de augmented Dickey-Fuller test (ADF) en de Phillips-Perron test (PP). Deze zullen we uiteindelijk gebruiken om de data te toetsen. Allereerst zal worden bekeken hoe deze testen werken en welke wiskundige principes hier achter zitten.

Opmerking 2.1. *In de praktijk blijkt dat d bijna nooit groter is dan 2. Elke keer als d wordt verhoogd, wordt namelijk ook de variantie van het proces verhoogd. Bekijk een MA(1) proces met $X_t = (1 - \theta B)a_t$. Dan is*

$$\Delta X_t = (1 - B)(1 - \theta B)a_t = (1 - (1 + \theta)B + \theta B^2)a_t = (1 - \theta_1 B - \theta_2 B^2)a_t$$

Dus $\theta_1 = 1 + \theta$ en $\theta_2 = -\theta$. Dus $\theta_1 + \theta_2 = 1$. Dus het model is niet inverteerbaar en een AR(∞)-representatie bestaat niet en dit schatten geeft problemen.

Merk nog op dat

$$V(x) = \gamma_0(X_t) = (1 + \theta^2)\sigma^2$$

$$\Delta X_t = w_t; V(w) = \gamma_0(w) = (1 + (1 + \theta)^2 + \theta^2)\sigma^2 = 2(1 + \theta + \theta^2)\sigma^2$$

$$V(w) - V(X) = (1 + \theta)^2\sigma^2 > 0$$

2.0.1 Dickey-Fuller test

Beschouw een AR(1) proces: $X_t = \phi X_{t-1} + a_t, 1 \leq t \leq T$ waarbij X_1, X_2, \dots, X_n zijn waargenomen met $a_t \sim NID(0, \sigma^2)$ en a_t onafhankelijk verdeeld ten

opzichte van $X_{t-1}, X_{t-2}, \dots, X_1$. Zij $X_0 = 0$.

Dan is $X_t | X_{t-1}, X_{t-2}, \dots, X_1 \sim N(\phi X_{t-1}, \sigma^2)$.

De OLS schatter van ϕ is dan te bepalen als volgt:

$$S(\phi) = \sum_{t=1}^T (X_t - \phi X_{t-1})^2$$

$$\frac{\partial S}{\partial \phi} = -2 \sum_{t=1}^T X_{t-1} (X_t - \phi X_{t-1}) = 0$$

$$\text{Geeft } \sum_{t=1}^T X_t X_{t-1} = \hat{\phi}_T \sum_{t=1}^T X_{t-1}^2$$

$$\hat{\phi}_T = \frac{\sum_{t=1}^T X_t X_{t-1}}{\sum_{t=1}^T X_{t-1}^2}$$

Dan volgt verder dat

$$\begin{aligned} (\hat{\phi}_T - \phi) &= \frac{\sum_{t=1}^T X_t X_{t-1}}{\sum_{t=1}^T X_{t-1}^2} - \phi = \frac{\sum_{t=1}^T (\phi X_{t-1} + a_t) X_{t-1}}{\sum_{t=1}^T X_{t-1}^2} - \phi \\ &= \frac{\sum_{t=1}^T X_{t-1} a_t}{\sum_{t=1}^T X_{t-1}^2} \end{aligned}$$

Als dan geldt dat $|\phi| < 1$, dan volgt uit Hamilton:

$$\sqrt{T}(\hat{\phi}_T - \phi) \stackrel{a}{\sim} N(0, \sigma^2 E(X_{t-1}^2))$$

$$X_{t-1} = a_{t-1}(1 - \phi B)^{-1} = a_{t-1}(1 + \phi B + \phi^2 B^2 + \dots) = a_{t-1} \sum_{i=0}^{\infty} \phi^i B^i$$

En $E(X_{t-1}) = E(X_t)$, dus

$$\begin{aligned} E(X_{t-1}^2) &= E\left(\left(\sum_{i=0}^{\infty} \phi^i a_{t-i}\right)^2\right) = E(a_t^2) + \phi^2 E(a_{t-1}^2) + \phi^4 E(a_{t-2}^2) + \dots \\ &= \sigma^2 \sum_{i=0}^{\infty} \phi^{2i} = \frac{\sigma^2}{1 - \phi^2} \end{aligned}$$

$$\text{Dus } \sqrt{T}(\hat{\phi}_T - \phi) \stackrel{a}{\sim} N(0, (1 - \phi^2))$$

Als $\phi = 1$ is er een probleem, want dan is de variantie 0, dus bekijk deze situatie apart:

$$(\hat{\phi}_T - 1) = \frac{\sum_{t=1}^T X_{t-1} a_t}{\sum_{t=1}^T X_{t-1}^2}$$

Dan volgt verder dat $X_t - X_{t-1} = a_t$, dus volgt dat $X_t = \sum_{s=1}^t a_s$, dus $X_t \sim N(0, \sigma^2 t)$. Verder is

$$X_t^2 = (X_{t-1} + a_t)^2 = X_{t-1}^2 + 2X_{t-1}a_t + a_t^2$$

$$X_{t-1}a_t = \frac{1}{2}(X_t^2 - X_{t-1}^2 - a_t^2)$$

$$\sum_{t=1}^T X_{t-1}a_t = \frac{1}{2} \sum_{t=1}^T (X_t^2 - X_{t-1}^2) - \frac{1}{2} \sum_{t=1}^T a_t^2 = \frac{1}{2}(x_T^2 - x_0^2) - \frac{1}{2} \sum_{t=1}^T a_t^2$$

Kies $x_0 = 0$, dan volgt dat

$$\frac{1}{\sigma^2 T} \sum_{t=1}^T X_{t-1}a_t = \frac{1}{2} \left(\frac{x_T}{\sigma\sqrt{T}} \right)^2 - \frac{1}{2\sigma^2} \frac{1}{T} \sum_{t=1}^T a_t^2$$

$$X_t \sim N(0, \sigma^2 t), \text{ dus } \frac{x_T}{\sigma\sqrt{T}} \sim N(0, 1), \text{ dus } \left(\frac{x_T}{\sigma\sqrt{T}} \right)^2 \sim \chi_1^2$$

$$E\left(\frac{1}{T} \sum_{t=1}^T a_t^2\right) = \frac{1}{T}(E(a_1^2) + E(a_2^2) + \dots + E(a_T^2)) = \frac{1}{T}(T \cdot \sigma^2) = \sigma^2$$

$$\text{Dus } T^{-1} \sum_{t=1}^T X_{t-1}a_t = \frac{1}{2}\sigma^2(X - 1), \text{ met } X \sim \chi_1^2$$

$$E(X_t^2) = V(X_t) + (E(X_t))^2 = \sigma^2 t$$

$$E(T^{-2} \sum_{t=1}^T X_{t-1}^2) = \frac{\sigma^2}{T^2} \sum_{t=1}^T (t-1) = \frac{\sigma^2}{2} \left(1 - \frac{1}{T}\right)$$

Dus volgt dat

$$T(\hat{\phi}_T - 1) = \frac{T^{-1} \sum_{t=1}^T X_{t-1}a_t}{T^{-2} \sum_{t=1}^T X_{t-1}^2}$$

Beschouw $\frac{X_T}{\sigma\sqrt{T}}$ als volgt. Functie: $[0, T] \rightarrow [0, 1], R_T(r) \mapsto X_{[rT]}(r)/\sigma\sqrt{T} = X_{t-1}/\sigma\sqrt{T}$, waarbij $[rT]$ het gehele deel van rT is ($r \in [0, 1]$).

Dus zo wordt $[0, 1]$ in $T + 1$ delen opgedeeld voor $r = 0, \frac{1}{T}, \dots, 1$ en $R_T(r)$ is dan een trapfunctie. Als $T \rightarrow \infty$, dan wordt $R_T(r)$ zwak continu en convergeert dat naar een Wiener Proces (Brownian Motion). [8]

Notatie 2.1. Dit Wiener Proces wordt genoteerd als $W(r)$.

Definitie 2.1 (Wiener Proces). Het Wiener Proces, ook wel Brownian Motion, is een continu stochastisch proces met onafhankelijke, normaal verdeelde aangroeiingen met de volgende eigenschappen:

1. $W_0 = 0$

2. De paden $t \rightarrow W_t$ zijn bijna zeker continu.
3. W_t heeft onafhankelijk aangroeiingen, dus voor $0 \leq s_1 < t_1 \leq s_2 < t_2$ geldt dat $W_{t_1} - W_{s_1}$ en $W_{t_2} - W_{s_2}$ onderling onafhankelijk zijn verdeeld.
4. Voor $0 \leq s < t$ is $W_t - W_s$ is $N(0, t - s)$ verdeeld. [2]

Verder geldt nog dat

$$W(1) \sim N(0, 1)$$

$$\sigma \cdot W(r) \sim N(0, \sigma^2 r)$$

$$\frac{[W(r)]^2}{r} \sim \chi_1^2$$

[8]

Dan volgt dat

$$\begin{aligned} T^{-2} \sum_{t=1}^T X_{t-1}^2 &= \frac{\sigma^2}{T} \sum_{t=1}^T \left(\frac{X_{t-1}}{\sigma\sqrt{T}} \right)^2 = \sigma^2 \sum_{t=1}^T T^{-1} \left(R_T \left(\frac{t-1}{T} \right) \right)^2 \\ &= \sigma^2 \sum_{t=1}^T \int_{(t-1)/T}^{t/T} [R_T(r)]^2 dr = \sigma^2 \int_0^1 [R_T(r)]^2 dr \Rightarrow \sigma^2 \int_0^1 [W(r)]^2 dr \end{aligned}$$

Verder is

$$\frac{1}{\sigma^2 T} \sum_{t=1}^T X_{t-1} a_t \sim^a \frac{1}{2} \sigma^2 (X - 1) \Rightarrow \frac{1}{2} \sigma^2 ([W(1)]^2 - 1)$$

$$T(\hat{\phi}_T - 1) \Rightarrow \frac{\frac{1}{2}([W(1)]^2 - 1)}{\int_0^1 [W(r)]^2 dr}$$

De noemer is positief, en de teller is natuurlijk $\frac{1}{2}([W(1)]^2 - 1)$ en die is negatief als $[W(1)]^2 = \chi_1^2 < 1$ en er geldt dat $P(\chi_1^2 < 1) = 0.68$. Dus dan volgt dat voor $T \rightarrow \infty$ dat $P(T(\hat{\phi}_T - 1) < 0) = 0.68$. Als χ_1^2 groot is, dan wordt de noemer dat ook, dus is de verdeling van $T(\hat{\phi}_T - 1)$ scheef naar links is verdeeld. [9]

Definitie 2.2 (Ito's formule).

$$g(W(1)) - g(W(0)) = \int_0^1 g'(W(r)) dW(r) + \frac{1}{2} \int_0^1 g''(W(r)) dr$$

[3] Als nu $g(W_r) = W(r)^2$, dan volgt dat

$$W(1)^2 - W(0)^2 = W(1)^2 = \int_0^1 2W(r) dW(r) + \frac{1}{2} \int_0^1 2 dr$$

$$\begin{aligned}
W(1)^2 &= 2 \int_0^1 W r dW(r) + 1 \\
\frac{1}{2}(W(1)^2 - 1) &= \int_0^1 W(r) dW(r) \\
\text{Dus } T(\hat{\phi}_T - 1) &\Rightarrow \frac{\int_0^1 W(r) dW(r)}{\int_0^1 [W(r)]^2 dr} \\
\tau &= \frac{T^{-1} \sum_{t=1}^T X_{t-1} a_t}{s_t (T^{-2} \sum_{t=1}^T X_{t-1}^2)^{\frac{1}{2}}} \Rightarrow \frac{\frac{1}{2} \sigma^2 (W(1)^2 - 1)}{\sigma \cdot (\sigma^2 \int_0^1 [W(r)]^2 dr)^{\frac{1}{2}}} \\
&= \frac{\frac{1}{2} ([W(1)]^2 - 1)}{\int_0^1 [W(r)]^2 dr)^{\frac{1}{2}}} = \frac{\int_0^1 W(r) dW(r)}{\int_0^1 [W(r)]^2 dr} \\
\sqrt{T}(\hat{\phi}_T - 1) &= \frac{T^{-\frac{3}{2}} \sum_{t=1}^T X_{t-1} a_t}{T^{-2} \sum_{t=1}^T X_{t-1}^2}
\end{aligned}$$

Als $T(\hat{\phi}_T - 1)$ een eindig iets is, dan gaat $\sqrt{T}(\hat{\phi}_T - 1)$ naar 0.

Definitie 2.3 (Dickey-Fuller). *De test gegeven door τ noemen we de Dickey-Fuller test.*

Definitie 2.4 (augmented Dickey-Fuller test). *Zo is er ook een test voor het geval dat het model een "trend" heeft, dit kijkt naar de volgende test:*

$$\tau_\mu = \frac{\hat{\phi}_T - 1}{se(\hat{\phi}_T)}$$

Definitie 2.5 (Phillips Perron test). *Deze test gebruikt het volgende*

$$\begin{aligned}
Z(\tau_\mu) &= \tau_\mu (\hat{\sigma}^2 / \hat{\sigma}_{SI}^2) - 1/2 (\hat{\sigma}_{SI}^2 - \hat{\sigma}^2) T [\hat{\sigma}_{SI}^2 \sum_{t=2}^T (X_{t-1} - \bar{x}_{-1})^2]^{-\frac{1}{2}} \\
\hat{x}_{-1} &= (T-1)^{-1} \sum_{t=1}^{T-1} x_t
\end{aligned}$$

2.0.2 Data, deel 3

Deze testen kunnen worden gebruikt om de orde van d uit ARIMA(p, d, q)-model te bepalen. Dat kan worden gedaan met de scripts die te vinden zijn op pagina 46, .1.4.

Wat we hier nu exact doen, is de data bekijken voor $d = 0$ en voor $d = 1$. Dit doen we eerste met de Augmented Dickey-Fuller Test (ADF) en daarna

met de Phillips-Perron Unit Root Test (PP).

De nulhypothese stelt dat de set stationair is. Als we dus een kleine p-waarde vinden, dan kunnen we aannemen dat de nulhypothese waar is, anders verwerpen we deze.

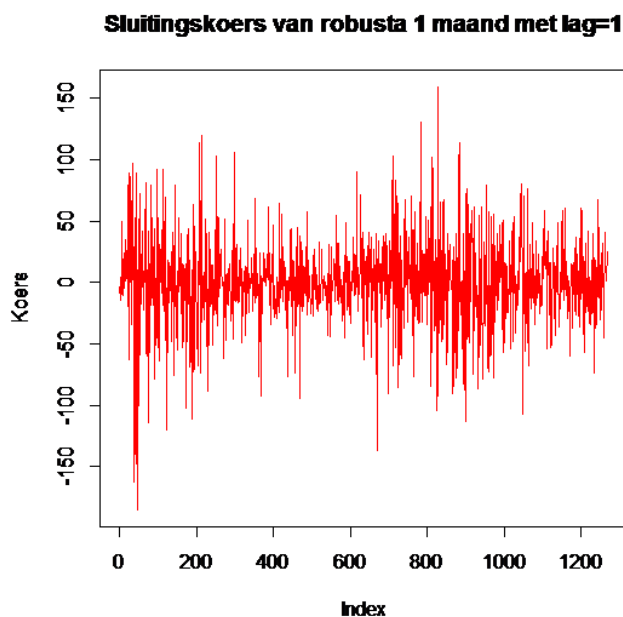
Voor $d = 0$ vinden we met de ADF dat de p-waarde 0.6141 is, dus dan verwerpen we de nulhypothese.

Voor $d = 1$ vinden we met de ADF dat de p-waarde 0.01 is, dus dan nemen we aan dat de set stationair is.

Voor $d = 0$ vinden we met de PP dat de p-waarde 0.6214 is, dus dan verwerpen we de nulhypothese.

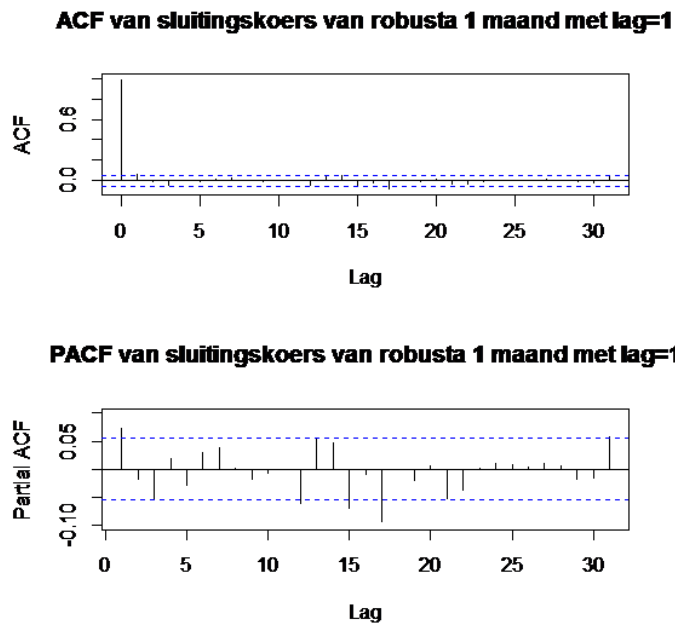
Voor $d = 1$ vinden we met de PP dat de p-waarde 0.01 is, dus dan nemen we aan dat de set stationair is.

Als we dan gaan kijken naar hoe de set eruit ziet voor $d = 1$, verkrijgen we de volgende plot:



Figuur 2.1: Plot van de robusta set met lag=1 (dus van $X_t - X_{t-1}$), 2e kolom

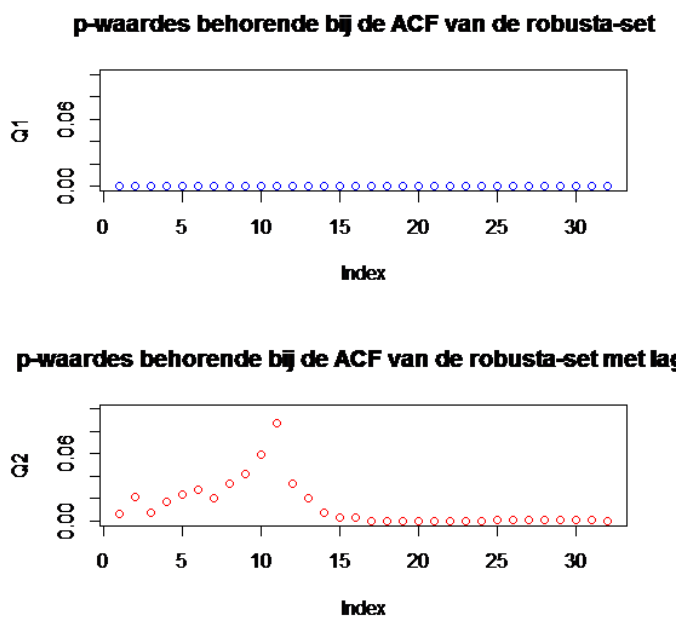
Daarbij hoort het volgende plaatje voor de ACF en de PACF, dat kan worden verkregen met het script te vinden op pagina 47, .1.5 :



Figuur 2.2: Plot van de acf en pacf van de robusta set met lag=1 (dus van $X_t - X_{t-1}$), 2e kolom

Nu kan met de Ljung-Box test worden bepaald of de returns white noise zijn of niet.

Daarvoor kan het script worden gebruikt dat is te vinden op pagina 47,.1.6, waarbij de verkregen p-waardes met de Ljung-Box test dan af te lezen zijn in de plot op de volgende pagina:



Figuur 2.3: Plot van de p waarden van de robusta set, bovenste zonder lag, tweede met lag=1 (dus van $X_t - X_{t-1}$), 2e kolom

2.0.3 Data, deel 4

Op basis hiervan kan worden gezocht naar een model. Allereerst kunnen verschillende testen worden gebruikt om de mogelijke orde van het model te bepalen. Dat kan worden gedaan met het script te vinden op pagina 48, .1.7. Dit geeft eigenlijk direct dat we alleen hoeven te kijken naar een model waarbij het $ARIMA(p, d, q)$ -model is met $d = 1$, $p = 3, 12, 17, 31$ en q gaan we dan ophogen om het beste model te kiezen. Dan vinde we dat elke keer de beste waarde te vinden is voor $q = 1$.

Er hoeft dus alleen te worden gekeken naar $ARIMA(3,1,q)$ -model, $ARIMA(12,1,q)$ -model, $ARIMA(17,1,q)$ -model en $ARIMA(31,1,q)$ -model. Nu kunnen er met behulp van het script, dat te vinden is op pagina 49, .1.8, functies worden aangemaakt om de AIC, de gecorrigeerde AIC en de BIC te bepalen.

Dan kan er een for-loop aangemaakt worden die alles afgaat en de matrix vult met de AIC, AICc en de BIC-waardes. Dit script is te vinden op pagina 49, .1.9.

Opmerking 2.2. *Er kan geen ARIMA(31,1,8)-model en ARIMA(31,1,9)-model worden gemaakt. Om toch te zorgen dat de matrix makkelijk kan worden doorzocht voor de beste waardes, kan het script worden gebruikt dat te vinden is op pagina 50, .1.10. Hierbij is ook handmatig de matrix op de plaats van het ARIMA(0,1,0)-model gevuld.*

De beste AIC, AICc en BIC kan worden gevonden met het script dat te vinden is op pagina 51, .1.11. Deze geeft dan dat de AIC en de AICc een ARIMA(17,1,4)-model kiest en dat de BIC een ARIMA(0,1,1)-model kiest.

2.0.4 Eerste conclusie

Op basis hiervan kan worden besloten om verder te gaan met het ARIMA(17,1,4)-model, dus hier is als criteria de AIC en AICc gekozen. De reden hiervoor is dat de BIC extra parameters enorm afstraft, terwijl dat de AIC en AICc dit minder doet. Hier kan natuurlijk over worden getwijfeld.

Hoofdstuk 3

GARCH-modellen

In dit hoofdstuk zullen martingalen worden beschreven en met behulp daarvan kunnen de GARCH-modellen met bijbehorende eigenschappen worden geïntroduceerd. Daarna zullen verschillende variaties op deze GARCH-modellen worden bekeken.

3.0.5 Martingaal

Definitie 3.1 (Martingaal). Een martingaal is een stochastisch proces $\{X_t\}$ met

(i) $E(|X_t|) < \infty$ voor alle t

(ii) $E(X_t | \mathfrak{S}_s) = x_s$ als $s \leq t$ met \mathfrak{S}_s de σ -algebra van gebeurtenissen bepaald door observaties in $[0, t]$ zodat $\mathfrak{S}_s \subset \mathfrak{S}_t$ als $s \leq t$.

3.0.6 ARCH(1)

Definitie 3.2 (ARCH(1)). Zij $\sigma_t = \mathfrak{S}(X_{t-1}) = (\alpha_0 + \alpha_1(X_{t-1} - \mu)^2)^{\frac{1}{2}}$ met $\alpha_0 > 0$ en $\alpha_1 \geq 0$.

Zij nu $U_t \sim NID(0, 1)$ en onafhankelijk van σ_t . Dan is $X_t = \mu + U_t \sigma_t$ white noise en dit is dan voorwaardelijk normaal verdeeld als volgt:

$X_t | X_{t-1}, X_{t-2}, \dots \sim NID(\mu, \sigma_t^2)$, zodanig dat $V(X_t | X_{t-1}) = \sigma_t^2 = \alpha_0 + \alpha_1(X_{t-1} - \mu)^2$

Als dan $\alpha_1 < 1$ dan volgt dat $V(X_t) = \frac{\alpha_0}{1 - \alpha_1}$ en dan bestaat er een X_t zwak stationaire oplossing.

Zij nu $\varepsilon_t = X_t - \mu = U_t \sigma_t$, zodanig dat ARCH(1) het volgende wordt:

$$\varepsilon_t | X_{t-1}, X_{t-2}, \dots \sim NID(0, \sigma_t^2)$$

$$\sigma_t^2 = \alpha_0 + \alpha_1 \varepsilon_{t-1}^2$$

Zij nu $v_t = \varepsilon_t^2 - \sigma_t^2$, dan is

$$\varepsilon_t^2 = \alpha_0 + \alpha_1 \varepsilon_{t-1}^2 + v_t$$

Omdat geldt dat $E(v_t | X_{t-1}, X_{t-2}, \dots) = 0$, is dit te bekijken als een AR(1)-proces, waarbij ε_t^2 de innovaties zijn. De heteroskedasticiteit zit dat in $\text{Var}(X_t | X_{t-1})$ en die is dus conditioneel, vandaar dat het een ARCH-model is.

Opmerking 3.1. *Heteroskedastisch betekent dat de variantie van deze errors variabel is.*

Definitie 3.3 (Kurtosis).

$$\kappa = \frac{E(X_t - \mu)^4}{E((X_t - \mu)^2)^2} - 3$$

Met behulp van deze kurtosis, κ , kan de verdeling van een proces worden vergeleken met een normale verdeling.

Als $\kappa > 0$, dan is er een smalle, hoge piek met dikke staarten, dit heet dan leptokurtosisch.

Als $\kappa = 0$, dan is dit een normale verdeling, dit heet dan mesokurtosisch.

Als $\kappa < 0$, dan is er een brede, lage piek en zijn er dunnere staarten, dit heet dan platykurtosisch.

[4]

Kurtosis van ARCH(1)

$$E(X_t - \mu)^4 = E(\varepsilon_t)^4 = E(U_t^4 \sigma_t^4) = E(U_t^4) E(\sigma_t^4)$$

Opmerking 3.2.

$$\begin{aligned} E(U_t^4) &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} x^4 \cdot \exp\left(-\frac{x^2}{2}\right) dx = \frac{1}{\sqrt{2\pi}} [x^3 \exp\left(-\frac{x^2}{2}\right)]_{-\infty}^{\infty} + \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} 3x^2 \cdot \exp\left(-\frac{x^2}{2}\right) dx \\ &= \frac{3}{\sqrt{2\pi}} \cdot E(x^2) = 3 \end{aligned}$$

$$\begin{aligned} &= 3E(\sigma_t^4) = 3E((\alpha_0 + \alpha_1 \varepsilon_{t-1}^2)^2) = 3E(\alpha_0^2 + 2\alpha_0 \alpha_1 E(\varepsilon_{t-1}^2) + \alpha_1^2 E(\varepsilon_{t-1}^4)) \\ &= 3\alpha_0^2 + 6\alpha_0 \alpha_1 E(\varepsilon_{t-1}^2) + 3\alpha_1^2 E(\varepsilon_{t-1}^4) \end{aligned}$$

$$\begin{aligned} \mu_4 &= E(\varepsilon_t^4) \text{ en } E(\varepsilon_t^2) = \frac{\alpha_0}{1 - \alpha_1} \text{ dus} \\ \mu_4(1 - 3\alpha_1^2) &= 3\alpha_0^2 + 6\alpha_0\alpha_1 E(\varepsilon_{t-1}^2) = 3\alpha_0^2(1 + \frac{2\alpha_1}{1 - \alpha_1}) \\ \mu_4 &= \frac{3\alpha_0^2(1 + \frac{2\alpha_1}{1 - \alpha_1})}{1 - 3\alpha_1^2} = \frac{3\alpha_0^2(1 + \alpha_1)}{(1 - \alpha_1)(1 - 3\alpha_1^2)} \\ (E(X_t - \mu)^2)^2 &= E(\varepsilon_t^2)^2 = E(U_t^2 \sigma_t^2)^2 = E(\sigma_t^2)^2 = \frac{\alpha_0^2}{(1 - \alpha_1)^2} \\ \kappa &= \frac{E(X_t - \mu)^4}{E((X_t - \mu)^2)^2} - 3 = \frac{\frac{3\alpha_0^2(1 + \alpha_1)}{(1 - \alpha_1)(1 - 3\alpha_1^2)}}{\frac{\alpha_0^2}{(1 - \alpha_1)^2}} - 3 = 3 \frac{(1 + \alpha_1)(1 - \alpha_1)}{1 - 3\alpha_1^2} - 3 = 3 \frac{1 - \alpha_1^2}{1 - 3\alpha_1^2} - 3 > 0 \end{aligned}$$

Dus er kan worden vastgesteld dat een ARCH(1)-proces een positieve kurtosis heeft, dus de verdeling is dan leptokurtotisch.

3.0.7 Data, deel 5

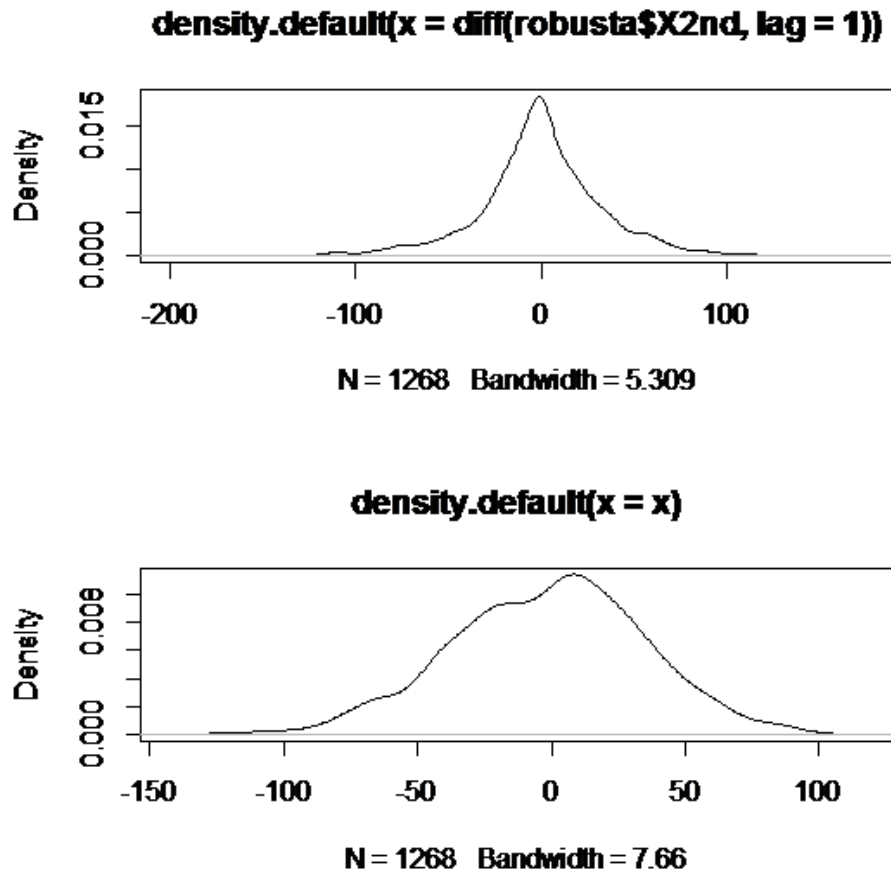
Zo kan er ook van de dataset de kurtosis worden bepaald. Dat kan worden gedaan met het script dat te vinden is op pagina 51, .1.12. Tevens is dit ook bekeken naar de robusta set met $d = 0$. (d is hier de d uit het ARIMA(p, d, q)-model, waarbij $d = 0$ de normale reeks betekent en $d = 1$ de reeks waarbij er naar de verschillen tussen de opvolgende data wordt gekeken).

We vinden hier dat de kurtosis van de verdeling met $d = 0$ gelijk is aan -1.026642 en dat de kurtosis van de verdeling met $d = 1$ gelijk is aan 2.433519 .

Opvallend is dus dat de verdeling met $d = 0$ een negatieve kurtosis heeft, wat niet bij tijdreeksen past. Vandaar dat het gebruiken van $d = 1$ een stuk beter lijkt uit te vallen, omdat deze een positieve kurtosis heeft en dit wel bij tijdreeksen past.

Verder is er nog gekeken naar de vorm van de tijdreeks. Allereerst is daarvoor het script gebruikt dat te vinden is op pagina 67.

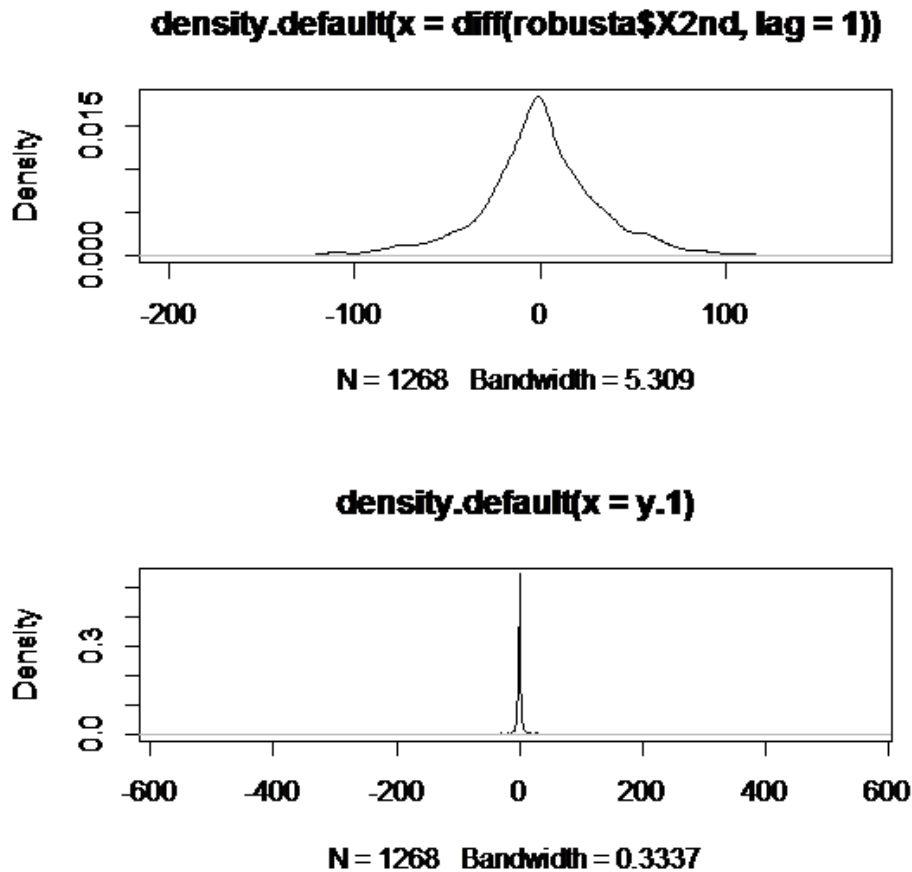
Hiermee is gekeken naar de dichtheidsfunctie van de tijdreeks van de returns, waarbij de lag gelijk is aan 1, het bijbehorende plaatje is te vinden op pagina 27, figuur 3.1. Dit is geplot ten opzichte van een normaalverdeling.



Figuur 3.1: Plot van de dichtheidsfunctie tov een simulatie van een normaalverdeling met $n=1269$

Hieruit is te concluderen dat de robustaset niet normaal verdeeld is, dit ziet er heel anders uit.

Zo is ook te zien in de volgende plot, dat de tijdreeks niet t-verdeeld is:



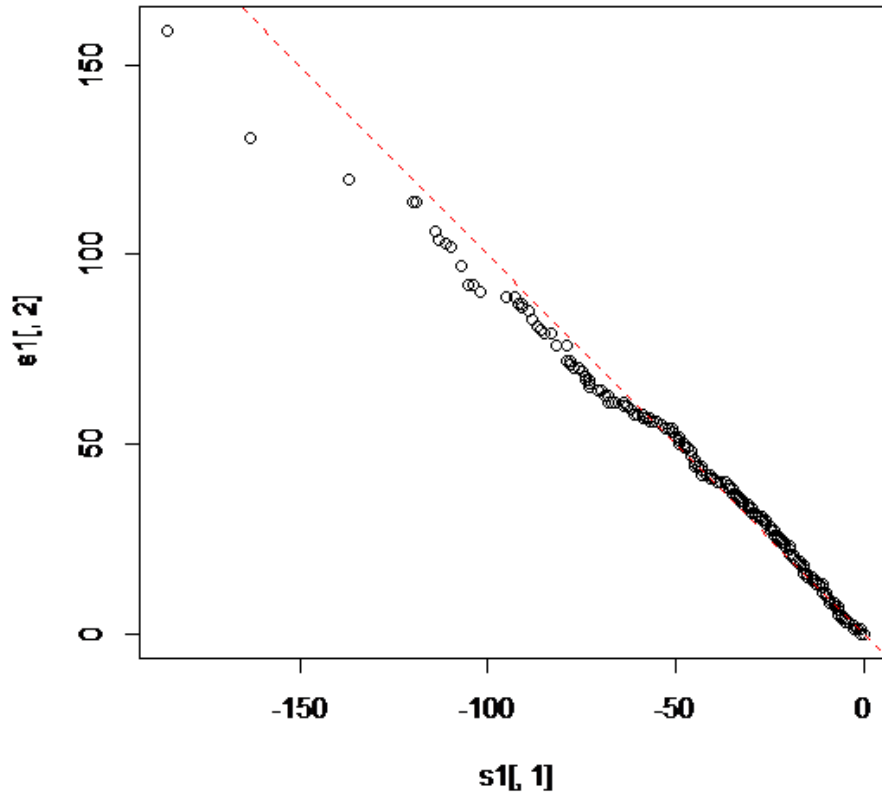
Figuur 3.2: Plot van de dichtheidsfunctie tov een t-verdeling met $df=1$

Aangezien het dus wat lastiger gaat worden, is het handig eerst te kijken naar de gegevens die er bekend zijn over de reeks, die met behulp van R kunnen worden gevonden. Dit is gedaan met het script dat te vinden is op pagina 67.

Daarin staan de lengte van de set, de verwachtingswaarde, standaarddeviatie, het maximum, minimum, de range, de scheefheid en de kurtosis.

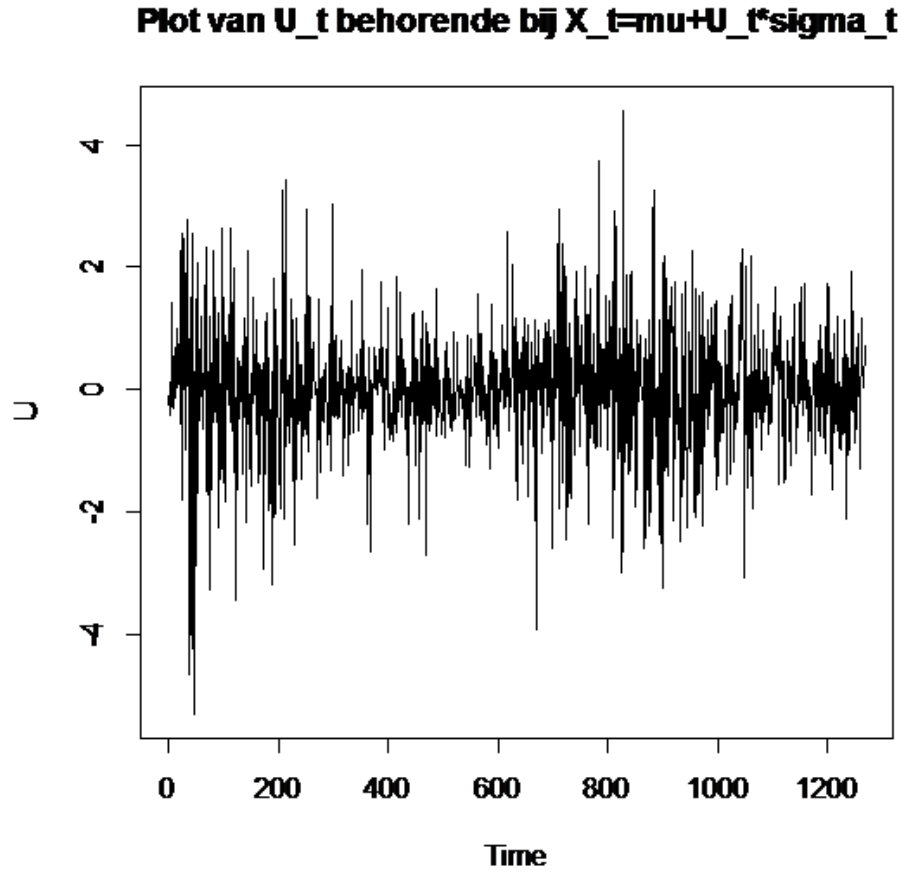
Als er dan wordt gekeken naar de symmetrie van de set, is te vinden dat dit in de staarten niet zo is. Dit is te zien in de plot op de volgende pagina:

Hieruit is te concluderen dat de set een lastige vorm heeft, die moeilijk te beschrijven is.



Figuur 3.3: Plot van de gesorteerde functie om de symmetrie te bekijken

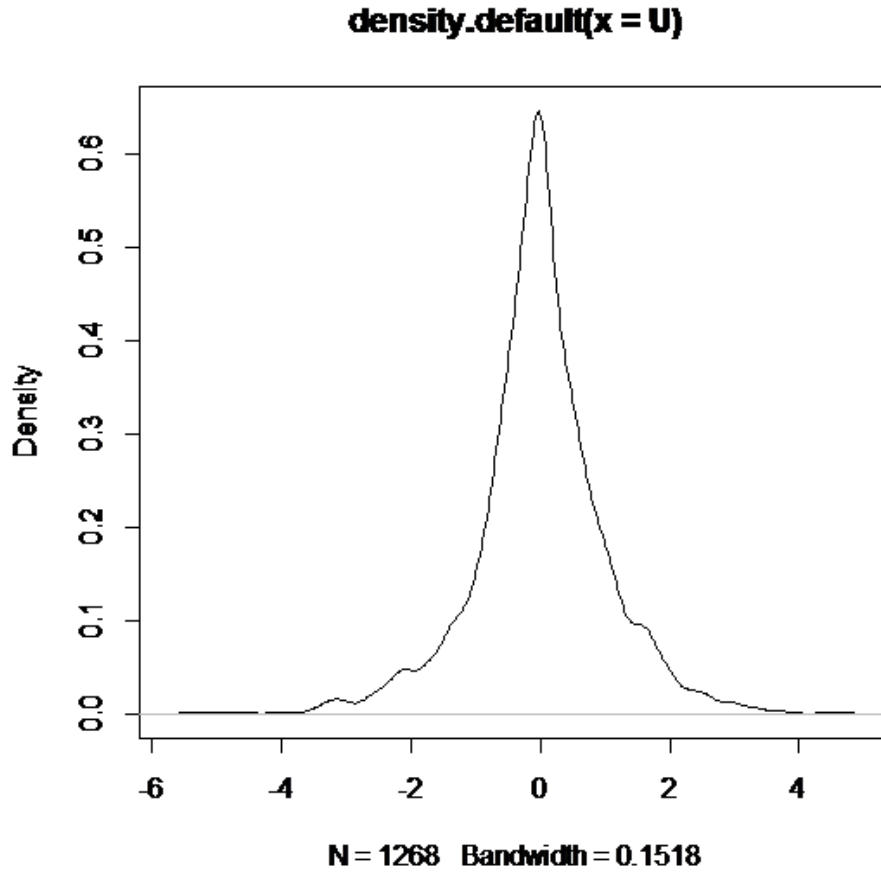
Daarom is het nog een optie om te kijken naar de vorm van de tijdsreeks die behoort bij het ARCH-proces, waarbij we ervan uitgaan dat geldt dat $X_t = \mu + U_t \cdot \sigma_t$, waarbij $E(X_t) = \mu$ en $\sigma_t^2 = E(X_t - \mu)^2$. Verder geldt daarbij dat $E(U_t) = 0$ en $V(U_t) = 1$. Dit is gedaan met het script dat is te vinden op pagina 67. De plot van de tijdsreeks ziet er dan als volgt uit:



Figuur 3.4: Plot van U_t behorende bij $X_t = \mu + U_t \cdot \sigma_t$

Daarvoor geldt dat inderdaad dat $E(U_t) = 0$ en $V(U_t) = 1$.

Als we nu kijken naar de dichtheidsfunctie van U_t (figuur 3.5), zien we dat voor deze verdeling geldt dat dit ook niet normaal verdeeld of t-verdeeld is en dat komt doordat de piek zo hoog is, met een dichtheid van meer dan 0.6. Daarom kunnen we deze niet verder beschrijven, maar wel concluderen dat dit een verdeling is die beter te beschrijven is. Wat het meest zou voldoen is dan een t-verdeling met veel vrijheidsgraden. Die hoge dichtheid van de piek wordt dan alleen niet bereikt.



Figuur 3.5: Plot van de dichtheidsfunctie van U_t , met U_t behorende bij $X_t = \mu + U_t \cdot \sigma_t$

Variante van innovaties van ARCH(1)

$$\begin{aligned}
 V(\varepsilon_t^2) &= E(\varepsilon_t^4) - (E(\varepsilon_t^2))^2 = \frac{3\alpha_0^2(1+\alpha_1)}{(1-3\alpha_1^2)(1-\alpha_1)} - \frac{\alpha_0^2}{(1-\alpha_1)^2} \\
 &= \frac{3\alpha_0^2(1-\alpha_1^2) - \alpha_0^2(1-3\alpha_1^2)}{(1-3\alpha_1^2)(1-\alpha_1)^2} = \frac{\alpha_0^2(3-3\alpha_1^2-1+3\alpha_1^2)}{(1-3\alpha_1^2)(1-\alpha_1)^2} \\
 &= 2 \frac{\alpha_0^2}{(1-3\alpha_1^2)(1-\alpha_1)^2} = \left(\frac{\alpha_0}{1-\alpha_1}\right)^2 \cdot \frac{2}{1-3\alpha_1^2} = \frac{2\hat{\sigma}^2}{1-3\alpha_1^2}
 \end{aligned}$$

3.0.8 ARCH(q)

$$\mathfrak{S}(X_{t-1}, X_{t-2}, \dots, X_{t-q}) = \left(\alpha_0 + \sum_{i=1}^q \alpha_i (X_{t-i} - \mu)^2 \right)^{\frac{1}{2}}$$

met $\alpha_0 > 0$ en $\alpha_i \geq 0$ voor $1 \leq i \leq q$.

Er is een zwak stationaire oplossing als $\sum_{i=1}^q \alpha_i < 1$.

Dan is $V(X_t) = \frac{\alpha_0}{1 - \sum_{i=1}^q \alpha_i}$

De functie wordt dan $\sigma_t^2 = \alpha_0 + \sum_{i=1}^q \alpha_i \varepsilon_{t-i}^2$

oftewel $\varepsilon_t^2 = \alpha_0 + \alpha(B) \varepsilon_{t-1}^2 + v_t$

3.0.9 GARCH(p,q)

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^q \alpha_i \varepsilon_{t-i}^2 + \sum_{i=1}^p \beta_i \sigma_{t-i}^2 = \alpha_0 + \alpha(B) \varepsilon_{t-1}^2 + \beta(B) + \sigma_t^2$$

met $p > 0, \beta_i > 0$ voor $1 \leq i \leq p$. Ook wel

$$\varepsilon_t^2 = \alpha_0 + (\alpha(B) + \beta(B)) \varepsilon_{t-1}^2 + v_t - \beta(B) v_{t-1}$$

zodanig dat $\varepsilon_t^2 \sim \text{ARMA}(m, p)$ met $m = \max(p, q)$.

Er is een zwak stationaire oplossing als $\alpha(B) + \beta(B) < 1$.

[10]

3.0.10 Modificaties van GARCH

De modellen die met R kunnen worden bekeken zijn:

- sGARCH (standaard GARCH-model)
- iGARCH (integrated GARCH-model)
- apARCH (asymmetric power ARCH-model)

Dan zijn er ook nog de modellen uit de fGARCH (family GARCH) klasse:

- TGARCH (treshold GARCH-model)
- GJR-GARCH (GJR-GARCH-model)
- AVGARCH (absolute value GARCH-model)
- NAGARCH (Nonlinear Asymmetric GARCH-model)
- NGARCH (Nonlinear GARCH-model)

Opmerking 3.3 (EGARCH). *Dat het eGARCH-model hier niet bij zit, is vrij logisch aangezien hier wordt uitgegaan van een logistische functie, wat deze tijdreeks duidelijk niet is. Uitproberen van dit model geeft ook direct aan dat dit model niet aan de tijdreeks kan worden gefit.*

Al deze modellen staan hieronder kort uitgewerkt. Daarbij wordt iedere keer van het volgende uitgegaan:

$$\varepsilon_t = x_t - \mu = U_t \sigma_t \text{ met } U_t \sim NID(0,1) \text{ en } U_t \text{ is onafhankelijk van } \sigma_t$$

$$\varepsilon_t | x_{t-1}, x_{t-2}, \dots \sim NID(0, \sigma_t^2)$$

$$v_t = \varepsilon_t^2 - \sigma_t^2$$

sGARCH

Een sGARCH(p, q) model is het standaard GARCH-model dat als eerder is beschreven, te vinden in paragraaf 3.0.9 op pagina 32.

iGARCH

Een iGARCH(p, q) model is precies het bovenstaande sGARCH model, maar dan met de extra voorwaarde dat $\sum_{i=1}^q \alpha_i + \sum_{j=1}^p \beta_j = 1$.

apARCH

Het apARCH(p, q) model ziet er als volgt uit:

$$\sigma_t^\delta = \alpha_0 + \sum_{i=1}^q \alpha_i (|\varepsilon_{t-i}| - \gamma_i \varepsilon_{t-i})^\delta + \sum_{j=1}^p \beta_j \sigma_{t-j}^\delta$$

met $\alpha_0 > 0, \alpha_i \geq 0, 1 \leq i \leq q, \beta_j \geq 0, 1 \leq j \leq p, \delta \geq 0$. [7]

eGARCH

Het eGARCH(p, q) model ziet er als volgt uit:

$$\log(\sigma_t^2) = \alpha_0 + \sum_{i=1}^q \alpha_i f\left(\frac{\varepsilon_{t-i}}{\sigma_{t-i}}\right) + \sum_{j=1}^p \beta_j \log(\sigma_{t-j}^2)$$

$$\text{met } f\left(\frac{\varepsilon_{t-i}}{\sigma_{t-i}}\right) = \theta \frac{\varepsilon_{t-i}}{\sigma_{t-i}} + \left(\left| \frac{\varepsilon_{t-i}}{\sigma_{t-i}} \right| - E\left(\left| \frac{\varepsilon_{t-i}}{\sigma_{t-i}} \right| \right) \right)$$

[10]

GJRGARCH

Het GJRGARCH model is te verkrijgen door het apARCH model gebruiken met $\delta = 2$.

TGARCH

Het TGARCH model is te verkrijgen door het apARCH model gebruiken met $\delta = 1$.

AVGARCH

Het AVGARCH(p, q) model ziet er als volgt uit:

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^p \alpha_i (|\varepsilon_{t-i} + b| - c(\varepsilon_{t-i} + b))^2 + \sum_{j=1}^q \beta_j \sigma_{t-j}^2$$

met $\alpha_0 > 0, \alpha_i \geq 0, 1 \leq i \leq p, \beta_j \geq 0, 1 \leq j \leq q$.

[5]

NGARCH of NAGARCH

Het NGARCH(p, q) of NAGARCH(p, q) model ziet er als volgt uit:

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^q \alpha_i (\varepsilon_{t-i} - \theta \sigma_{t-i})^2 + \sum_{j=1}^p \beta_j \sigma_{t-j}^2$$

met $\alpha_0 > 0, \alpha_i \geq 0, 1 \leq i \leq q, \beta_j \geq 0, 1 \leq j \leq p, \sigma \in \mathbf{R}$.

[1]

3.0.11 Data, deel 6

Al deze modellen kunnen worden gefit aan de robusta-set, waarbij $d=1$, met een ARMA(17,4)-model om de verwachtingswaarde te simuleren.

Daarvoor kunnen de scripts op pagina 51, .1.13 gebruikt worden.

Uiteindelijk zijn er ook waardes die niet konden worden bepaald, deze kunnen handmatig worden gevuld. Dat is gegaan op dezelfde manier als eerder met de ARMA-modellen.

3.0.12 Data, deel 7

Daarna kan er met het script op pagina 57, .1.14 per matrix de beste AIC, AICc en BIC waarden worden bepaald. Die zijn te vinden in onderstaande tabel, met bijbehorend model. Wat hier direct aan opvalt is dat de AIC en AICc telkens hetzelfde model selecteren. Verder valt het op dat de BIC ons telkens adviseert het (1,1) model te kiezen (bij apARCH (1)). Nu is dus de vraag wat het eindcriterium gaat worden. Omdat er in de literatuur geen duidelijkheid wordt gegeven over welk criterium nu het beste is, zal de AIC (en AICc) aan worden gehouden, dit is eerder ook al gedaan namelijk. Het beste model volgens zowel de AIC en AICc zou het ARMA(17,4)-iGARCH(4,8) model zijn. De BIC geeft dat het beste model het ARMA(17,4)-GJRGARCH(1,1) model zou zijn.

Type model	Model	AIC	Model	AICc	Model	BIC
sGARCH	(2,7)	12450.13	(2,7)	12451.73	(1,1)	12579.54
iGARCH	(4,8)	12417.03	(4,8)	12418.96	(1,1)	12585.72
apARCH	(1)	12571.52	(1)	12572.41	(1)	12689.86
TGARCH	(1,1)	12441.37	(1,1)	12443.09	(1,1)	12575.81
AVGARCH	(8,2)	12450.21	(8,2)	12452.03	(1,1)	12576.94
NAGARCH	(8,3)	12455.84	(8,3)	12457.45	(1,1)	12584.65
NGARCH	(3,6)	12455.84	(3,6)	12457.45	(1,1)	12584.65
GJRGARCH	(1,1)	12446.64	(1,1)	12447.61	(1,1)	12570.13

3.0.13 Data, deel 9

Dit hele proces kan ook herhaald worden voor de 2, 3, en 4 maanden contracten. Dat kan worden gedaan met de scripts te vinden in het apart bijgevoegde bestandje, extrascripts.txt . Dat geeft de volgende uitslagen:

Data :	X _{3rd}					
p-waardes						
ADF	lag=0	0.6283	ADF	lag=1	< 0.01	
PP	lag=0	0.6277	PP	lag=1	0.01	
Box.test						
lag=0	Geen white noise		lag=1	Geen white noise		
AR.ordes						
	1	12	17	31		
Type model	Model	AIC	Model	AICc	Model	BIC
ARMA	(17,3)	12519.08	(17,3)	12519.82	(0,1)	12534.17
sGARCH	(2,4)	12348.50	(2,4)	12349.72	(1,1)	12467.26
iGARCH	(2,1)	12333.20	(2,1)	12334.16	(2,1)	12456.68
apARCH	(1)	12489.27	(1)	12490.09	(1)	12602.47
TGARCH	(1,1)	12350.61	(1,1)	12351.49	(1,1)	12468.95
AVGARCH	(7,6)	12316.61	(7,6)	12318.54	(1,1)	12458.30
NAGARCH	(6,8)	12326.71	(6,8)	12328.76	(1,1)	12463.74
NGARCH	(1,1)	12353.60	(1,1)	12354.49	(1,1)	12471.94
GJRGARCH	(6,4)	12350.86	(6,4)	12352.46	(1,1)	12474.41

Data :	X ₄ th					
p-waardes						
ADF	lag=0	0.6638	ADF	lag=1	< 0.01	
PP	lag=0	0.6542	PP	lag=1	0.01	
Box.test						
lag=0	Geen white noise		lag=1	Geen white noise		
AR.ordes						
	1	12	31			
Type model	Model	AIC	Model	AICc	Model	BIC
ARMA	(0,1)	12391.51	(0,1)	12391.52	(0,0)	12400.36
sGARCH	(2,2)	12214.79	(2,2)	12214.86	(1,1)	12235.50
iGARCH	(2,8)	12217.28	(2,8)	12217.53	(1,1)	12241.02
apARCH	(1)	12387.58	(1)	12387.60	(1)	12403.01
TGARCH	(1,1)	12214.72	(1,1)	12214.75	(1,1)	12235.30
AVGARCH	(8,3)	12195.94	(8,3)	12196.23	(2,1)	12229.04
NAGARCH	(4,4)	12200.31	(4,4)	12200.48	(1,1)	12228.80
NGARCH	(1,1)	12214.67	(1,1)	12214.70	(1,1)	12235.25
GJRGARCH	(1,1)	12212.09	(1,1)	12212.12	(1,1)	12232.67

Data :	X ₅ th					
p-waardes						
ADF	lag=0	0.6743	ADF	lag=1	< 0.01	
PP	lag=0	0.6683	PP	lag=1	0.01	
Box.test						
lag=0	Geen white noise		lag=1	Geen white noise		
AR.ordes						
	1	12	31			
Type model	Model	AIC	Model	AICc	Model	BIC
ARMA	(0,1)	12348.89	(0,1)	12348.90	(0,0)	12357
sGARCH	(1,1)	12160.37	(1,1)	12160.41	(1,1)	12180.96
iGARCH	(7,8)	12136.23	(7,8)	12136.72	(1,1)	12186.49
apARCH	(1)	12345.94	(1)	12345.96	(1)	12361.38
TGARCH	(1,1)	12157.64	(1,1)	12157.67	(1,1)	12178.22
AVGARCH	(8,4)	12145.41	(8,4)	12145.74	(1,1)	12173.14
NAGARCH	(3,4)	12146.72	(3,4)	12146.86	(1,1)	12173.78
NGARCH	(1,1)	12159.35	(1,1)	12159.38	(1,1)	12179.93
GJRGARCH	(1,1)	12157.79	(1,1)	12157.82	(1,1)	12178.37

3.1 Mogelijkheden

Er zijn verder naast deze modellen ook nog vele andere modellen om dit te modelleren en te kijken of dit past. Dit deel is dus ook meer als suggesties te beschouwen, wanneer men mogelijk beter passende modellen zou wensen. Hieronder staan een aantal multivariate GARCH-modellen uitgewerkt. Het principe hiervan is dat de modellen met meerdere tegelijk worden bekeken door middel van het bekijken van de covariantie tussen de verschillende datasets.

3.1.1 VEC specificatie van GARCH(p,q)

Definitie 3.4 (VEC). Zij de VEC operator de operator die het volgende doet op een $(m \times n)$ matrix A .

$$A = \begin{pmatrix} a_{11} & - & - & - & a_{1n} \\ | & | & | & | & | \\ | & | & | & | & | \\ | & | & | & | & | \\ a_{m1} & - & - & - & a_{mn} \end{pmatrix} \text{vec}(A) = (a_{11}, a_{21}, \dots, a_{m1}, a_{12}, \dots, a_{m2}, \dots, a_{1n}, \dots, a_{mn})^T$$

Definitie 3.5 (VECH). Als nu A symmetrisch is, dan geeft $\text{vec}(A)$ dubbele info. Definieer daarom de vech operator als volgt:

Zij A een $(n \times n)$ matrix, dan gebruikt $\text{vech}(A)$ alle info onder de diagonaal.

$$A = \begin{pmatrix} a_{11} & - & - & - & a_{1n} \\ | & | & | & | & | \\ | & | & | & | & | \\ | & | & | & | & | \\ a_{m1} & - & - & - & a_{mn} \end{pmatrix} \text{vech}(A) = (a_{11}, a_{21}, \dots, a_{n1}, a_{22}, \dots, a_{n2}, \dots, a_{n-1,n-1}, a_{n,n-1}, a_{nn})^T$$

Opmerking 3.4. Zo wordt een $(n \times n)$ matrix getransformeerd naar een vector van grootte $n^* = \frac{n(n+1)}{2}$.

Zij nu H_t een $(d \times d)$ positieve definitieve matrix en zij ε_t een d -dimensionale innovatie met $E(\varepsilon_t) = 0$ en de covariantie matrix van ε_t gegeven door H_t zodanig dat

$$\varepsilon_t = H_t^{\frac{1}{2}} \zeta_t$$

waarbij $E(\zeta_t) = 0$ en de covariantie matrix van $\zeta_t = I_d$, waarbij I_d een $(d \times d)$ identiteitsmatrix is.

Opmerking 3.5. d is hier telkens het aantal datasets dat wordt bekeken.

We definiëren nu $d^* = \frac{d(d+1)}{2}$.

Zij nu $h_t = \text{vech}(H_t)$ en zij $\eta_t = \text{vech}(\varepsilon_t \varepsilon_t^T)$.

Dan is de VEC specificatie van het multivariate GARCH(p, q) model als volgt:

$$h_t = \omega + \sum_{i=1}^q A_i \eta_{t-i} + \sum_{j=1}^p B_j h_{t-j}$$

waarbij A_i en B_j de parameter matrices zijn met $(d^*)^2$ parameters en ω met d^* parameters.

3.1.2 BEKK

Zij C_0 een symmetrische matrix waarbij we alleen kijken naar de hoek onder de diagonaal.

Zij verder A_{ki} en B_{kj} ($d \times d$) parameter matrices.

Dan is de BEKK specificatie als volgt:

$$H_t = C_0 C_0^T + \sum_{k=1}^K \sum_{i=1}^q A_{ki}^T \varepsilon_{t-i} \varepsilon_{t-i}^T A_{ki} + \sum_{k=1}^K \sum_{j=1}^p B_{kj}^T H_{t-j} B_{kj}$$

Opmerking 3.6. BEKK geeft dan dat H_t een positieve definitie matrix is.

[6]

3.1.3 CCC

Deze doet wat meer denken aan een gegeneraliseerd GARCH model en ziet er als volgt uit:

$$\sigma_{it}^2 = \omega_i + \sum_{j=1}^q \alpha_{i,j} \varepsilon_{i,t-j}^2 + \sum_{j=1}^p \beta_{i,j} \sigma_{i,t-j}^2, i = 1, \dots, n$$

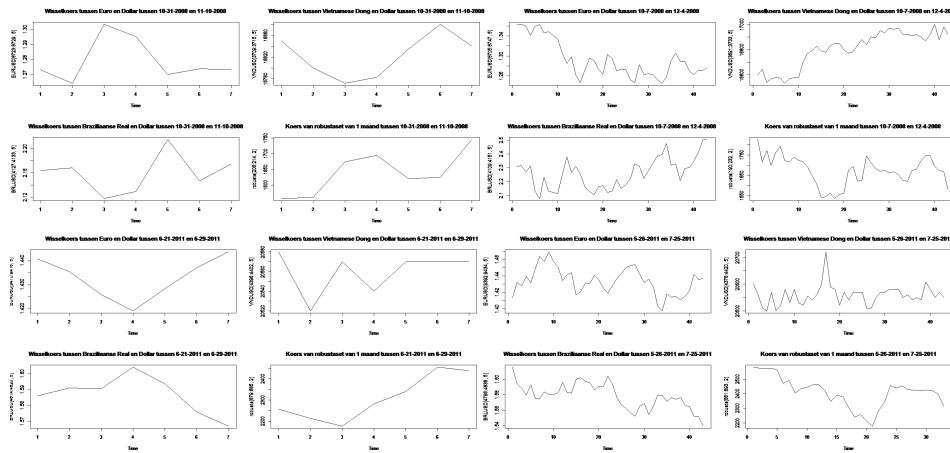
$$\sigma_{ij,t} = \rho_{ij} \sigma_{it} \sigma_{jt}, i, j = 1, \dots, n, i \neq j$$

[11]

3.1.4 Verklarend gedrag en theorie rond sprongen

Op basis van de dataset is het mogelijk nog te kijken naar delen van de data waar de sprongen erg groot zijn. Om dat te doen, is het nodig de data eerst te "filteren". Dat kan met het script op pagina 59, .1.16.

Met behulp hiervan is de data handmatig bekeken, wanneer er meer dan 2 data met sprongen groter dan 100 (naar beneden of naar boven) zijn geweest in een klein tijdsbestek (zeg minder dan 4 dagen), dan is deze data geselecteerd en is daarnaar gekeken. Tegelijkertijd is op datzelfde interval gekeken naar de wisselkoersen van de Braziliaanse, Vietnamese en Europese valuta ten opzichte van de dollar.



Figuur 3.6: Plot van alle wisselkoersen waarbij het verschil groter dan 100 is in de robustaset voor een periode van minimaal 2 dagen

De bijbehorende plots zijn hier te bekijken, dat zijn de figuren 3.6 en 3.7.

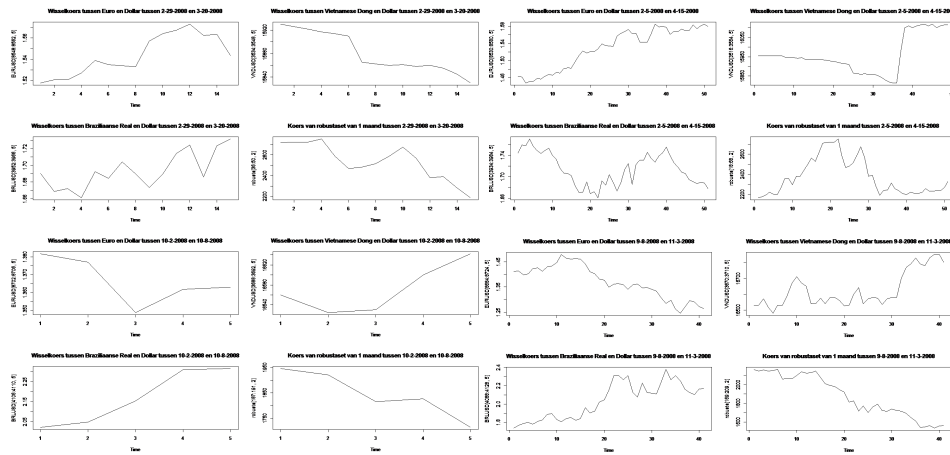
De data rond deze plots is daarna bekeken op basis van de variantie per tijdseenheid. Deze is ook genomen op een omgeving van 18 dagen eerder en 18 dagen later (Dit getal is arbitrair te noemen). Ditzelfde is ook gedaan voor de wisselkoersen op diezelfde omgeving.

Daarna is gekeken naar de verhoudingen tussen de koersen op diezelfde intervals, waarbij dus een deling is gedaan tussen de verhouding van het kleine tijdsinterval ten opzichte van het grotere interval. De gebruikte scripts en bijbehorende verhoudingen zijn allemaal te vinden op 59, .1.16.

Deze verhoudingen van de koersen zijn daarna allemaal vermenigvuldigd en op basis hiervan is het volgende vermoeden gebaseerd:

Als we dan kijken naar het punt waarop deze koers start en op welke waarde deze eindigt, zien we dat de verschillen in de koers klein zijn, respectievelijk 120, 60, 120 en 200. De bijbehorende tijdseenheden zijn dan (10,10,20,15). Terwijl dat bij de bekeken koers de verschillen respectievelijk 500,250,200,250 zijn in kleine tijdseenheden (14,4,6,6).

Nu is hieruit op te maken dat wanneer de verhouding tussen de vari-



Figuur 3.7: Plot van alle wisselkoersen waarbij het verschil groter dan 100 is in de robuustset voor een periode van minimaal 2 dagen

anties op het kleine interval ten opzichte van het grote interval als een vermenigvuldiging van alle verhoudingen van zowel de eurUSD, de vnUSD en de brUSD kleiner is dan 1 (arbitrair getal), dan kan er worden verwacht dat er sprongen van de koers zijn die in principe zowel omhoog als omlaag gaan springen rond een evenwichtsstand, terwijl dat wanneer de vermenigvuldiging groter is dan 1, dan zal de koers enorme sprongen gaan ondervinden, naar danwel boven, danwel beneden en dus wegschieten van dat beginpunt.

Oftewel, er zit een mooi verband tussen de wisselkoersen en de beurs, namelijk dat wanneer er (met een netto vermenigvuldiging) minder risico is ten opzichte van een vorige periode bij de wisselkoersen, dan kan dit ook worden verwacht voor de koers van de koffie, waarbij wel weer kan worden verwacht dat er sprongen op en neer kunnen zijn.

Wanneer er juist een groter risico bij de wisselkoersen komt, dan zal dit zich tevens uiteten op de koffiemarkt als een groter risico.

Verder is het nog opvallend dat als er een echt groot getal als verhouding is van een bepaalde koers, in dit geval die van de Braziliaanse Real tussen 2008-10-2 en 2008-10-8 op basis van 2008-9-8 en 2008-11-3, dan zien we ook direct enorme veranderingen plaatsvinden in een korte tijd. Hier maar liefst 2 keer 100 in 4 dagen tijd.

Hieruit kan de conclusie worden getrokken dat wanneer varianties grote veranderingen ondergaan, dat er dan ook een navolging is in de koers op een korte termijn. Deze verschillen zijn dan ineens erg groot.

Verder is in de figuren 3.6,3.7 op pagina 41, 42 te zien dat de robustakoers een mooie navolging is ten opzichte van de andere koersen.

Opmerking 3.7. *Dit lijkt alleen te gelden voor echt grote sprongen, dit is verder niet geverifieerd met behulp van andere data. Vandaar dat het een vermoeden blijft.*

3.1.5 Eindconclusie

Op basis van de datasets is te concluderen dat er allereerst moet worden gekeken naar een 'integrated' model, wat inhoudt dat er wordt gekeken naar een model met een lag, die hoogstwaarschijnlijk gelijk is aan 1. Dus er wordt dan gekeken naar de returns.

Deze returns zijn dan te bekijken op verschillende manieren. Allereerst is het opvallend dat de vorm van de tijdreeks niet netjes zoals bij een normaal of t is verdeeld, maar dat deze lastig te beschrijven is. Verder is ook te zien dat de pieken die zich afspelen juist een indicatie afgeven dat een GARCH model of een variatie hierop erg passend zou kunnen zijn. Deze variaties zijn allemaal te bekijken en de uiteindelijke conclusie is dat we per dataset (er zijn er 4 die zijn bekeken) een meest passend model hebben gevonden. Een korte opsomming daarvan:

- Voor de dataset van de *robustaX2nd.txt* file, is gevonden dat het ARMA(17,4)-iGARCH(4,8) model het beste zal passen.
- Voor de dataset van de *robustaX3rd.txt* file, is gevonden dat het ARMA(17,3)-AVGARCH(7,6) model het beste zal passen.
- Voor de dataset van de *robustaX4th.txt* file, is gevonden dat het ARMA(0,1)-AVGARCH(8,3) model het beste zal passen.
- Voor de dataset van de *robustaX5th.txt* file, is gevonden dat het ARMA(0,1)-iGARCH(7,8) model het beste zal passen.

Overigens is hier dan elke keer gekozen op basis van de AIC, dat lijkt ook het meest logisch, omdat het interessant is om een passend model voor de

“toekomst” te vinden. Opvallend hieraan is natuurlijk dat de modellen erg verschillend lijken te zijn.

Verder zou het nog interessant kunnen zijn om te kijken naar multivariate garch modellen en te kijken of deze waardes geven die beter scoren dan de tot nu toe gevonden modellen.

Appendix

.1 Scripts

De gebruikte packages uit R zijn:

- forecast
- rugarch
- timeSeries
- TSA
- xlsx

.1.1 Script 1

Hieronder staat het script waarmee de file is ingelezen, de tijdreeks is geplot en de verwachtingswaarde en variantie zijn bepaald. Verder staat er ook het script waarmee de ACF en PACF zijn geplot.

```
robusta<-read.table("C:/Users/Alwin/Mijn documenten/Wiskunde  
/Tijdreeksen/Bachelorproject/robusta.txt")-
```

```
ts.plot(robusta$X2nd,xlab="Datum",ylab="Koers",  
main="Sluitingskoers van robusta 1 maand",col="blue")
```

```
mean(robusta$X2nd);var(robusta$X2nd)
```

```
[1] 1863.085
```

```
[1] 134660.5
```

```
par(mfrow=c(2,1))
```

```
acf(robusta$X2nd,main="ACF van sluitingskoers van robusta 1 maand")
```

```
pacf(robusta$X2nd,main="PACF van sluitingskoers van robusta 1 maand",ylim=c(-0.1,0.1))
```

.1.2 Script 2

Hieronder staat het script waarmee handmatig de Ljung-Box test is af te nemen.

```
acfrobusta <- acf(robusta$X2nd,main="ACF van sluitingskoers van robusta 1 maand")
Q<-matrix(NA,nrow=32,ncol=1)
Q[1] <- T*(T+2)*acfrobusta$acf[1]^2/(T-1)
for (i in 2:32){ Q[i]<-Q[i-1]+T*(T+2)*acfrobusta$acf[i]^2/(T-i)}
P <- matrix(NA,nrow=32,ncol=1)
for (i in 1:32){ P[i] <- 1 - pchisq(Q[i],df=i)}
```

.1.3 Script 3

Hieronder staat het script waarmee direct de Ljung-Box test is af te nemen.

```
Q1<-matrix(NA,nrow=32,ncol=1)
for (i in 1:32){
Q1[i] <- Box.test(robusta$X2nd,lag=i,type="Ljung-Box")$p.value
}
```

.1.4 Script 4

Hieronder staat het script waarmee de augmented Dickey-Fuller test en de Phillips-Perron test af te nemen zijn.

```
adf.test(robusta$X2nd)
```

Augmented Dickey-Fuller Test

```
data: robusta$X2nd
Dickey-Fuller = -1.9155, Lag order = 10, p-value = 0.6141
alternative hypothesis: stationary
```

```
adf.test(diff(robusta$X2nd,lag=1))
```

Augmented Dickey-Fuller Test

```
data: diff(robusta$X2nd, lag = 1)
Dickey-Fuller = -10.5592, Lag order = 10, p-value = 0.01
alternative hypothesis: stationary
```

Warning message:

```
In adf.test(diff(robusta$X2nd, lag = 1)) :
  p-value smaller than printed p-value
PP.test(robusta$X2nd)
```

Phillips-Perron Unit Root Test

```
data: robusta$X2nd
Dickey-Fuller = -1.8982, Truncation lag parameter = 7, p-value = 0.6214
```

```
PP.test(diff(robusta$X2nd,lag=1))
```

Phillips-Perron Unit Root Test

```
data: diff(robusta$X2nd, lag = 1)
Dickey-Fuller = -32.8745, Truncation lag parameter = 7, p-value = 0.01
```

.1.5 Script 5

Hieronder staat het script waarmee de tijdreeks van de robusta-set met $d = 1$ is te plotten met bijbehorende codes voor de plots van de ACF en de PACF.

```
ts.plot(diff(robusta$X2nd,lag=1),xlab="Index",ylab="Koers",
main="Sluitingskoers van robusta 1 maand met lag=1",col="red")
par(mfrow=c(2,1))
acf(diff(robusta$X2nd,lag=1),main="ACF van sluitingskoers van robusta 1 maand met 1
pacf(diff(robusta$X2nd,lag=1),main="PACF van sluitingskoers van robusta 1 maand met
```

.1.6 Script 6

Hieronder staat het script waarmee de Ljung-Box test kan worden uitgevoerd op de robusta-set met $d = 1$.

```

Q2<-matrix(NA,nrow=32,ncol=1)
for (i in 1:32){
Q2[i] <- Box.test(diff(robusta$X2nd,lag=1),lag=i,type="Ljung-Box")$p.value
}
par(mfrow=c(2,1))
plot(Q1,main="p-waardes behorende bij de ACF van de robusta-set",ylim=c(-0.1,0.1),col="red")
plot(Q2,main="p-waardes behorende bij de ACF van de robusta-set met lag=1",ylim=c(-0.1,0.1),col="red")

```

.1.7 Script 7

Hieronder staat het script waarmee het beste passende ARMA (ARIMA(p,1,q)) model kan worden bepaald. De gebruikte testen zijn de Burg, MLE, OLS en Yule-Walker test. Op deze testen zal niet verder worden ingegaan, omdat deze test uiteindelijk alleen het onderliggende model bepaald.

```

arburg1<-ar.burg(x=diff(robusta$X2nd,lag=1),aic=TRUE)
arburg2<-ar.burg(x=diff(robusta$X2nd,lag=1),aic=FALSE)
armle1<-ar.mle(x=diff(robusta$X2nd,lag=1),aic=TRUE)
armle2<-ar.mle(x=diff(robusta$X2nd,lag=1),aic=FALSE)
arols1<-ar.ols(x=diff(robusta$X2nd,lag=1),aic=TRUE)
arols2<-ar.ols(x=diff(robusta$X2nd,lag=1),aic=FALSE)
aryw1<-ar.yw(x=diff(robusta$X2nd,lag=1),aic=TRUE)
aryw2<-ar.yw(x=diff(robusta$X2nd,lag=1),aic=FALSE)

```

```

arburg1$order;arburg2$order
[1] 17
[1] 31
armle1$order;armle2$order
[1] 3
[1] 12
arols1$order;arols2$order
[1] 31
[1] 31
aryw1$order;aryw2$order
[1] 17
[1] 31

```

.1.8 Script 8

Hieronder staat het script waarmee de functies voor de AIC, AICc en de BIC kunnen worden gemaakt.

```
aic<-function(loglik,p,q,n){
k <- p+q+1
theaic <- 2*k - 2*loglik
return(theaic)
}
```

```
aicc<-function(loglik,p,q,n){
k <- p+q+1
theaicc <- 2*k - 2*loglik + 2*k*(k+1)/(n-k-1)
return(theaicc)
}
```

```
bic<-function(loglik,p,q,n){
k <- p+q+1
thebic <- -2*loglik + k*log(n)
return(thebic)
}
```

.1.9 Script 9

Hieronder staat het script waarmee een matrix is gemaakt die gevuld is met alle modellen van de eerder genoemde ordes (te vinden in .1.7) en bijbehorende AIC, AICc en BIC (overigens heb ik de waarde van qmax bepaald door vooraf te zoeken en zo een maximum vast te stellen).

```
porde<- c(0, armle1$order, armle2$order, arburg1$order, arburg2$order)
qmax<- 10
Matrix <- matrix(NA,nrow=qmax*(length(porde)),ncol=7)

for (j in 1:5){
p=porde[j];
for (q in 1:qmax+1){
i=1;q=q-1;
teller=(q+1)+(qmax*(j-1));
```



```

n=length(robusta$X2nd)-i
myfit <- arima(x=robusta$X2nd,order=c(p,i,q),include.mean=F)
loglik <- myfit$loglik
theaic <- aic(myfit$loglik,p,q,n)
theaicc <- aicc(myfit$loglik,p,q,n)
thebic <- bic(myfit$loglik,p,q,n)
Matrix[teller,1]=p; Matrix[teller,2]=i; Matrix[teller,3]=q;
Matrix[teller,4]=loglik; Matrix[teller,5]=theaic;
Matrix[teller,6]=theaicc;Matrix[teller,7]=thebic;
}
}

```

.1.10 Script 10

Hieronder staat het script waarmee er wordt gezorgd dat er geen problemen ontstaan met het doorzoeken door de matrix door de log likelihood voor het ARIMA(3,1,8) model en ARIMA(3,1,9) model $-\infty$ te maken, zodat de AIC, AICc en de BIC heel groot worden. Tevens is hier handmatig de waarde van het ARIMA(0,1,0) model nog bepaald.

```

Matrix[49,1]=31;Matrix[50,1]=31;Matrix[49,2]=1
;Matrix[50,2]=1;Matrix[49,3]=8;Matrix[50,3]=9
Matrix[49,4]=-Inf;Matrix[50,4]=-Inf
Matrix[49,5]=Inf;Matrix[50,5]=Inf
Matrix[49,6]=Inf;Matrix[50,6]=Inf
Matrix[49,7]=Inf;Matrix[50,7]=Inf

```

```

myfit<-arima(x=robusta$X2nd,order=c(0,1,0),include.mean=F)
Matrix[1,1]=0;Matrix[1,2]=1;
Matrix[1,3]=0;Matrix[1,4]=myfit$loglik;

```

```

Matrix[1,5]=aic(Matrix[1,4],Matrix[1,1],Matrix[1,3],length(robusta$X2nd)-1)
Matrix[1,6]=aicc(Matrix[1,4],Matrix[1,1],Matrix[1,3],length(robusta$X2nd)-1)
Matrix[1,7]=bic(Matrix[1,4],Matrix[1,1],Matrix[1,3],length(robusta$X2nd)-1)

```

.1.11 Script 11

Hieronder staat het script waarmee de beste AIC, AICc en BIC waarden kunnen worden gevonden.

```
min(Matrix[,5])
[1] 12597.31
min(Matrix[,6])
[1] 12598.13
min(Matrix[,7])
[1] 12618.55
```

De eerste hoort bij een ARIMA(17,1,4)-model, de tweede ook en de derde hoort bij een ARIMA(0,1,1)-model.

.1.12 Script 12

Hieronder staat het script waarmee de kurtosis kan worden bepaald van de robusta-set met $d = 0$ en $d = 1$. Deze staan er beide in ter vergelijking.

```
kurtosis(robusta$X2nd) [1]
[1] -1.026642
kurtosis(diff(robusta$X2nd,lag=1)) [1]
[1] 2.433519
```

.1.13 Script 13

Hieronder staat het script dat gebruikt is om alle mogelijke type GARCH-modellen te fitten. qmax en pmax zijn door vooraf te zoeken vastgesteld. Het eerste script brengt een matrix voort met alle sGARCH modellen.

```
pmaxi=8;qmaxi=8
Matrix2.2nd.1.s=matrix(NA,nrow=(pmaxi+1)*qmaxi,ncol=6)

for (p in 1:(pmaxi+1)){
p=p-1;
for (q in 2:(qmaxi+1)){
q=q-1;
n=length(diff(robusta$X2nd,lag=1));
teller=q+p*qmaxi
```

```

spec=ugarchspec(variance.model=list(model="sGARCH",garchOrder=c(p,q)),
mean.model=list(armaOrder=c(17,4)));
fit=ugarchfit(spec=spec,data=diff(robusta$X2nd,lag=1));
loglik <- fit@fit$LLH;
theaic <- aic(loglik,p+17,q+4,n)
theaicc <- aicc(loglik,p+17,q+4,n)
thebic <- bic(loglik,p+17,q+4,n)
Matrix2.2nd.1.s[teller,1]=p;Matrix2.2nd.1.s[teller,2]=q;
Matrix2.2nd.1.s[teller,3]=loglik;Matrix2.2nd.1.s[teller,4]=theaic;
Matrix2.2nd.1.s[teller,5]=theaicc;Matrix2.2nd.1.s[teller,6]=thebic;
}
}

```

Het volgende script brengt een matrix voort met alle iGARCH modellen.

```

pmaxi=8;qmaxi=8
Matrix2.2nd.1.i=matrix(NA,nrow=(pmaxi+1)*qmaxi,ncol=6)

for (p in 1:(pmaxi+1)){
p=p-1;
for (q in 2:(qmaxi+1)){
q=q-1;
n=length(diff(robusta$X2nd,lag=1));
teller=q+p*qmaxi
spec=ugarchspec(variance.model=list(model="iGARCH",garchOrder=c(p,q)),
mean.model=list(armaOrder=c(17,4)));
fit=ugarchfit(spec=spec,data=diff(robusta$X2nd,lag=1));
loglik <- fit@fit$LLH;
theaic <- aic(loglik,p+17,q+4,n)
theaicc <- aicc(loglik,p+17,q+4,n)
thebic <- bic(loglik,p+17,q+4,n)
Matrix2.2nd.1.i[teller,1]=p;Matrix2.2nd.1.i[teller,2]=q;
Matrix2.2nd.1.i[teller,3]=loglik;Matrix2.2nd.1.i[teller,4]=theaic;
Matrix2.2nd.1.i[teller,5]=theaicc;Matrix2.2nd.1.i[teller,6]=thebic;
}
}

```

Het volgende script brengt een matrix voort met alle apARCH modellen.

```

qmaxi=6
Matrix2.2nd.1.ap=matrix(NA,nrow=qmaxi,ncol=6)

for (q in 2:(qmaxi+1)){
q=q-1;
n=length(diff(robusta$X2nd,lag=1));
teller=q
spec=ugarchspec(variance.model=list(model="apARCH",garchOrder=c(0,q)),
mean.model=list(armaOrder=c(17,4)));
fit=ugarchfit(spec=spec,data=diff(robusta$X2nd,lag=1));
loglik <- fit@fit$LLH;
theaic <- aic(loglik,17,q+4,n)
theaicc <- aicc(loglik,17,q+4,n)
thebic <- bic(loglik,17,q+4,n)
Matrix2.2nd.1.ap[teller,1]=0;Matrix2.2nd.1.ap[teller,2]=q;
Matrix2.2nd.1.ap[teller,3]=loglik;Matrix2.2nd.1.ap[teller,4]=theaic;
Matrix2.2nd.1.ap[teller,5]=theaicc;Matrix2.2nd.1.ap[teller,6]=thebic;
}

```

Het volgende script brengt een matrix voort met alle TGARCH modellen.

```

pmaxi=8;qmaxi=8
Matrix2.2nd.2.t=matrix(NA,nrow=(pmaxi+1)*qmaxi,ncol=6)

for (p in 1:(pmaxi+1)){
p=p-1;
for (q in 2:(qmaxi+1)){
q=q-1;
n=length(diff(robusta$X2nd,lag=1));
teller=q+p*qmaxi
spec=ugarchspec(variance.model=list(model="fGARCH",garchOrder=c(p,q),submodel="TGARCH"),
mean.model=list(armaOrder=c(17,4)));
fit=ugarchfit(spec=spec,data=diff(robusta$X2nd,lag=1));
loglik <- fit@fit$LLH;
theaic <- aic(loglik,p+17,q+4,n)
theaicc <- aicc(loglik,p+17,q+4,n)
thebic <- bic(loglik,p+17,q+4,n)
Matrix2.2nd.2.t[teller,1]=p;Matrix2.2nd.2.t[teller,2]=q;
}
}

```

```
Matrix2.2nd.2.t[teller,3]=loglik;Matrix2.2nd.2.t[teller,4]=theaic;
Matrix2.2nd.2.t[teller,5]=theaicc;Matrix2.2nd.2.t[teller,6]=thebic;
}
}
```

```
Matrix2.2nd.2.t[72,3]=-Inf;Matrix2.2nd.2.t[72,4]=Inf;
Matrix2.2nd.2.t[72,5]=Inf;Matrix2.2nd.2.t[72,6]=Inf;
```

Het volgende script brengt een matrix voort met alle AVGARCH modellen.

```
pmaxi=8;qmaxi=8
Matrix2.2nd.2.av=matrix(NA,nrow=(pmaxi+1)*qmaxi,ncol=6)

for (p in (pmaxi+1):(pmaxi+1)){
p=p-1;
for (q in (qmaxi):(qmaxi+1)){
q=q-1;
n=length(diff(robusta$X2nd,lag=1));
teller=q+p*qmaxi
spec=ugarchspec(variance.model=list(model="fGARCH",garchOrder=c(p,q),submodel="AVGAR
mean.model=list(armaOrder=c(17,4)));
fit=ugarchfit(spec=spec,data=diff(robusta$X2nd,lag=1));
loglik <- fit@fit$LLH;
theaic <- aic(loglik,p+17,q+4,n)
theaicc <- aicc(loglik,p+17,q+4,n)
thebic <- bic(loglik,p+17,q+4,n)
Matrix2.2nd.2.av[teller,1]=p;Matrix2.2nd.2.av[teller,2]=q;
Matrix2.2nd.2.av[teller,3]=loglik;Matrix2.2nd.2.av[teller,4]=theaic;
Matrix2.2nd.2.av[teller,5]=theaicc;Matrix2.2nd.2.av[teller,6]=thebic;
}
}
```

```
Matrix2.2nd.2.av[71,1]=8;Matrix2.2nd.2.av[71,2]=7
Matrix2.2nd.2.av[71,3]=-Inf;Matrix2.2nd.2.av[71,4]=Inf
Matrix2.2nd.2.av[71,5]=Inf;Matrix2.2nd.2.av[71,6]=Inf
```

Het volgende script brengt een matrix voort met alle NAGARCH modellen.

```

pmaxi=8;qmaxi=8
Matrix2.2nd.2.na=matrix(NA,nrow=(pmaxi+1)*qmaxi,ncol=6)

for (p in 1:(pmaxi+1)){
p=p-1;
for (q in 2:(qmaxi+1)){
q=q-1;
n=length(diff(robusta$X2nd,lag=1));
teller=q+p*qmaxi
spec=ugarchspec(variance.model=list(model="fGARCH",garchOrder=c(p,q),submodel="NAGAR")
mean.model=list(armaOrder=c(17,4)));
fit=ugarchfit(spec=spec,data=diff(robusta$X2nd,lag=1));
loglik <- fit@fit$LLH;
theaic <- aic(loglik,p+17,q+4,n)
theaicc <- aicc(loglik,p+17,q+4,n)
thebic <- bic(loglik,p+17,q+4,n)
Matrix2.2nd.2.na[teller,1]=p;Matrix2.2nd.2.na[teller,2]=q;
Matrix2.2nd.2.na[teller,3]=loglik;Matrix2.2nd.2.na[teller,4]=theaic;
Matrix2.2nd.2.na[teller,5]=theaicc;Matrix2.2nd.2.na[teller,6]=thebic;
}
}

```

```
Hulpna<-c(24,41,45,63,66)
```

```

for (i in 1:5){
Matrix2.2nd.2.na[Hulpna[i],3]==-Inf; Matrix2.2nd.2.na[Hulpna[i],4]=Inf
Matrix2.2nd.2.na[Hulpna[i],5]=Inf;Matrix2.2nd.2.na[Hulpna[i],6]=Inf;
}

```

Het volgende script brengt een matrix voort met alle NGARCH modellen.

```

pmaxi=8;qmaxi=8
Matrix2.2nd.2.n=matrix(NA,nrow=(pmaxi+1)*qmaxi,ncol=6)

for (p in 1:(pmaxi+1)){
p=p-1;
for (q in 2:(qmaxi+1)){
q=q-1;

```

```

n=length(diff(robusta$X2nd,lag=1));
teller=q+p*qmaxi
spec=ugarchspec(variance.model=list(model="fGARCH",garchOrder=c(p,q),submodel="NGAR
mean.model=list(armaOrder=c(17,4)));
fit=ugarchfit(spec=spec,data=diff(robusta$X2nd,lag=1));
loglik <- fit@fit$LLH;
theaic <- aic(loglik,p+17,q+4,n)
theaicc <- aicc(loglik,p+17,q+4,n)
thebic <- bic(loglik,p+17,q+4,n)
Matrix2.2nd.2.n[teller,1]=p;Matrix2.2nd.2.n[teller,2]=q;
Matrix2.2nd.2.n[teller,3]=loglik;Matrix2.2nd.2.n[teller,4]=theaic;
Matrix2.2nd.2.n[teller,5]=theaicc;Matrix2.2nd.2.n[teller,6]=thebic;
}
}

```

```
Matrix2.2nd.2.n[72,1]=8;Matrix2.2nd.2.n[72,2]=8
```

```
Hulpn<-c(2,72)
```

```

for (i in 1:2){
Matrix2.2nd.2.n[Hulpn[i],3]=-Inf; Matrix2.2nd.2.n[Hulpn[i],4]=Inf
Matrix2.2nd.2.n[Hulpn[i],5]=Inf;Matrix2.2nd.2.n[Hulpn[i],6]=Inf;
}

```

Het volgende script brengt een matrix voort met alle GJRGARCH modellen.

```

pmaxi=8;qmaxi=8
Matrix2.2nd.2.gjr=matrix(NA,nrow=(pmaxi+1)*qmaxi,ncol=6)

for (p in (pmaxi+1):(pmaxi+1)){
p=p-1;
for (q in (qmaxi+1):(qmaxi+1)){
q=q-1;
n=length(diff(robusta$X2nd,lag=1));
teller=q+p*qmaxi
spec=ugarchspec(variance.model=list(model="fGARCH",garchOrder=c(p,q),submodel=""GJR
mean.model=list(armaOrder=c(17,4)));

```

```

fit=ugarchfit(spec=spec,data=diff(robusta$X2nd,lag=1));
loglik <- fit@fit$LLH;
theaic <- aic(loglik,p+17,q+4,n)
theaicc <- aicc(loglik,p+17,q+4,n)
thebic <- bic(loglik,p+17,q+4,n)
Matrix2.2nd.2.gjr[teller,1]=p;Matrix2.2nd.2.gjr[teller,2]=q;
Matrix2.2nd.2.gjr[teller,3]=loglik;Matrix2.2nd.2.gjr[teller,4]=theaic;
Matrix2.2nd.2.gjr[teller,5]=theaicc;Matrix2.2nd.2.gjr[teller,6]=thebic;
}
}

```

```
Hulpgjr<-c(69,70,72)
```

```

for (i in 1:3){
Matrix2.2nd.2.gjr[Hulpgjr[i],3]==-Inf; Matrix2.2nd.2.gjr[Hulpgjr[i],4]=Inf
Matrix2.2nd.2.gjr[Hulpgjr[i],5]=Inf;Matrix2.2nd.2.gjr[Hulpgjr[i],6]=Inf;
}
}

```

.1.14 Script 14

Hieronder staat het script waarmee de beste AIC, AICc en BIC waarden kunnen worden bepaald. Dit is per matrix gedaan.

```

min(Matrix2.2nd.1.s[,4]);min(Matrix2.2nd.1.s[,5]);min(Matrix2.2nd.1.s[,6]);
[1] 12450.13
[1] 12451.73
[1] 12579.54

```

De AIC geeft ARMA(17,4)-sGARCH(2,7) De AICc geeft ARMA(17,4)-sGARCH(2,7)
De BIC geeft ARMA(17,4)-sGARCH(1,1)

```

min(Matrix2.2nd.1.i[,4]);min(Matrix2.2nd.1.i[,5]);min(Matrix2.2nd.1.i[,6]);
[1] 12417.03
[1] 12418.96
[1] 12585.72

```

De AIC geeft ARMA(17,4)-iGARCH(4,8) De AICc geeft ARMA(17,4)-iGARCH(4,8)
De BIC geeft ARMA(17,4)-iGARCH(1,1)


```
min(Matrix2.2nd.1.ap[,4]);min(Matrix2.2nd.1.ap[,5]);min(Matrix2.2nd.1.ap[,6]);
[1] 12571.52
[1] 12572.41
[1] 12689.86
```

De AIC geeft ARMA(17,4)-apARCH(1) De AICc geeft ARMA(17,4)-apARCH(1)
De BIC geeft ARMA(17,4)-apARCH(1)

```
min(Matrix2.2nd.2.t[,4]);min(Matrix2.2nd.2.t[,5]);min(Matrix2.2nd.2.t[,6]);
[1] 12461.62
[1] 12462.59
[1] 12585.11
```

De AIC geeft ARMA(17,4)-TGARCH(1,1) De AICc geeft ARMA(17,4)-TGARCH(1,1)
De BIC geeft ARMA(17,4)-TGARCH(1,1)

```
min(Matrix2.2nd.2.av[,4]);min(Matrix2.2nd.2.av[,5]);min(Matrix2.2nd.2.av[,6]);
[1] 12441.37
[1] 12443.09
[1] 12575.81
```

De AIC geeft ARMA(17,4)-AVGARCH(8,2) De AICc geeft ARMA(17,4)-
AVGARCH(8,2) De BIC geeft ARMA(17,4)-AVGARCH(1,1)

```
min(Matrix2.2nd.2.na[,4]);min(Matrix2.2nd.2.na[,5]);min(Matrix2.2nd.2.na[,6]);
[1] 12450.21
[1] 12452.03
[1] 12576.94
```

De AIC geeft ARMA(17,4)-NAGARCH(8,3) De AICc geeft ARMA(17,4)-
NAGARCH(8,3) De BIC geeft ARMA(17,4)-NAGARCH(1,1)

```
min(Matrix2.2nd.2.n[,4]);min(Matrix2.2nd.2.n[,5]);min(Matrix2.2nd.2.n[,6]);
[1] 12455.84
[1] 12457.45
[1] 12584.65
```

De AIC geeft ARMA(17,4)-NGARCH(3,6) De AICc geeft ARMA(17,4)-NGARCH(3,6)
De BIC geeft ARMA(17,4)-NGARCH(1,1)

```
min(Matrix2.2nd.2.gjr[,4]);min(Matrix2.2nd.2.gjr[,5]);min(Matrix2.2nd.2.gjr[,6]);
[1] 12446.64
[1] 12447.61
[1] 12570.13
```

De AIC geeft ARMA(17,4)-GJRGARCH(1,1) De AICc geeft ARMA(17,4)-GJRGARCH(1,1) De BIC geeft ARMA(17,4)-GJRGARCH(1,1)

.1.15 Script 15

Hieronder staat het script waarmee een Monte Carlo simulatie is gedaan met het ARMA(17,4)-iGARCH(4,8) model en het ARMA(17,4)-GJRGARCH(1,1) model.

```
spec.48i=ugarchspec(variance.model=list(model="iGARCH",garchOrder=c(4,8)),
mean.model=list(armaOrder=c(17,4)))
fit.48i=ugarchfit(spec=spec.48i,data=diff(robusta$X2nd,lag=1))
Matrix.48i<-matrix(NA,nrow=length(diff(robusta$X2nd,lag=1)),ncol=100)
for (i in 1:100){ sim.48i=ugarchsim(fit.48i,n.sim=length(diff(robusta$X2nd,lag=1)),
for (j in 1:1268){ Matrix.48i[j,i]=sim.48i@simulation$seriesSim[j]}}
```

.1.16 Script 16

Hieronder staat het script waarmee er is gezocht naar de data die een verklaring behoeft.

```
x.1 <- (diff(robusta$X2nd,lag=1)> 100)
y.1 <- (diff(robusta$X2nd,lag=1)< -100)
x.2 <- diff(robusta$X2nd,lag=1)[x]
y.2 <- diff(robusta$X2nd,lag=1)[y]
x.3 <- which(diff(robusta$X2nd,lag=1)> 100)
y.3 <- which(diff(robusta$X2nd,lag=1)< -100)
x.4<-robusta[x.3,1]
y.4<-robusta[y.3,1]
```

```
> plot.new()
> par(mfrow=c(2,2))
```

```
ts.plot(EURUSD[8548:8562,5],main="Wisselkoers tussen Euro en Dollar tussen 2-29-2000")
```

```
ts.plot(VNDUSD[3534:3548,5],main="Wisselkoers tussen Vietnamese Dong en Dollar tusse  
ts.plot(BRLUSD[3952:3966,5],main="Wisselkoers tussen Braziliaanse Real en Dollar tus  
ts.plot(robusta[36:50,2],main="Koers van robustaset van 1 maand tussen 2-29-2008 en
```

```
EURUSD[8530,1:5]
```

	Date..GMT.	Open	High	Low	Last
8530	2008-02-05	1.483	1.4834	1.462	1.4637

```
EURUSD[8580,1:5]
```

	Date..GMT.	Open	High	Low	Last
8580	2008-04-15	1.584	1.5875	1.575	1.5791

```
ts.plot(EURUSD[8530:8580,5],main="Wisselkoers tussen Euro en Dollar tussen 2-5-2008  
ts.plot(VNDUSD[3516:3564,5],main="Wisselkoers tussen Vietnamese Dong en Dollar tusse  
ts.plot(BRLUSD[3934:3984,5],main="Wisselkoers tussen Braziliaanse Real en Dollar tus  
ts.plot(robusta[18:68,2],main="Koers van robustaset van 1 maand tussen 2-5-2008 en 4
```

```
ts.plot(EURUSD[8702:8706,5],main="Wisselkoers tussen Euro en Dollar tussen 10-2-2008  
ts.plot(VNDUSD[3688:3692,5],main="Wisselkoers tussen Vietnamese Dong en Dollar tusse  
ts.plot(BRLUSD[4106:4110,5],main="Wisselkoers tussen Braziliaanse Real en Dollar tus  
ts.plot(robusta[187:191,2],main="Koers van robustaset van 1 maand tussen 10-2-2008 e
```

```
EURUSD[8684,1:5]
```

	Date..GMT.	Open	High	Low	Last
8684	2008-09-08	1.4383	1.4428	1.4053	1.4126

```
EURUSD[8724,1:5]
```

	Date..GMT.	Open	High	Low	Last
8724	2008-11-03	1.2748	1.2898	1.2594	1.2641

```
ts.plot(EURUSD[8684:8724,5],main="Wisselkoers tussen Euro en Dollar tussen 9-8-2008  
ts.plot(VNDUSD[3670:3710,5],main="Wisselkoers tussen Vietnamese Dong en Dollar tusse  
ts.plot(BRLUSD[4088:4128,5],main="Wisselkoers tussen Braziliaanse Real en Dollar tus  
ts.plot(robusta[169:209,2],main="Koers van robustaset van 1 maand tussen 9-8-2008 en
```

```
ts.plot(EURUSD[8723:8729,5],main="Wisselkoers tussen Euro en Dollar tussen 10-31-2008")
ts.plot(VNDUSD[3709:3715,5],main="Wisselkoers tussen Vietnamese Dong en Dollar tussen 10-31-2008")
ts.plot(BRLUSD[4127:4133,5],main="Wisselkoers tussen Braziliaanse Real en Dollar tussen 10-31-2008")
ts.plot(robusta[208:214,2],main="Koers van robustaset van 1 maand tussen 10-31-2008")
```

```
EURUSD[8705,1:5]
```

	Date..GMT.	Open	High	Low	Last
8705	2008-10-07	1.3484	1.3739	1.3479	1.3617

```
EURUSD[8747,1:5]
```

	Date..GMT.	Open	High	Low	Last
8747	2008-12-04	1.2701	1.2848	1.2548	1.277

```
ts.plot(EURUSD[8705:8747,5],main="Wisselkoers tussen Euro en Dollar tussen 10-7-2008")
ts.plot(VNDUSD[3691:3733,5],main="Wisselkoers tussen Vietnamese Dong en Dollar tussen 10-7-2008")
ts.plot(BRLUSD[4109:4151,5],main="Wisselkoers tussen Braziliaanse Real en Dollar tussen 10-7-2008")
ts.plot(robusta[190:232,2],main="Koers van robustaset van 1 maand tussen 10-7-2008")
```

```
ts.plot(EURUSD[9410:9416,5],main="Wisselkoers tussen Euro en Dollar tussen 6-21-2011")
ts.plot(VNDUSD[4396:4402,5],main="Wisselkoers tussen Vietnamese Dong en Dollar tussen 6-21-2011")
ts.plot(BRLUSD[4814:4820,5],main="Wisselkoers tussen Braziliaanse Real en Dollar tussen 6-21-2011")
ts.plot(robusta[879:885,2],main="Koers van robustaset van 1 maand tussen 6-21-2011")
```

```
EURUSD[9392,1:5]
```

	Date..GMT.	Open	High	Low	Last
9392	2011-05-26	1.4084	1.4206	1.4065	1.4138

```
EURUSD[9434,1:5]
```

	Date..GMT.	Open	High	Low	Last
9434	2011-07-25	1.4396	1.4406	1.4323	1.4374

```
ts.plot(EURUSD[9392:9434,5],main="Wisselkoers tussen Euro en Dollar tussen 5-26-2011")
ts.plot(VNDUSD[4378:4420,5],main="Wisselkoers tussen Vietnamese Dong en Dollar tussen 5-26-2011")
ts.plot(BRLUSD[4796:4838,5],main="Wisselkoers tussen Braziliaanse Real en Dollar tussen 5-26-2011")
```

```
ts.plot(robusta[861:893,2],main="Koers van robustaset van 1 maand tussen 5-26-2011 e
```

```
Koers<-matrix(NA,nrow=4,ncol=8)
```

```
Koers[1,1]<- var(robusta[36:50,2])/length(robusta[36:50,2]);  
Koers[2,1]<-var(robusta[187:191,2])/length(robusta[187:191,2]);  
Koers[3,1]<-var(robusta[208:214,2])/length(robusta[208:214,2]);  
Koers[4,1]<-var(robusta[879:885,2])/length(robusta[879:885,2]);
```

```
Koers[1,2]<-var(robusta[18:68,2])/length(robusta[18:68,2]);  
Koers[2,2]<-var(robusta[169:209,2])/length(robusta[169:209,2]);  
Koers[3,2]<-var(robusta[190:232,2])/length(robusta[190:232,2]);  
Koers[4,2]<-var(robusta[861:903,2])/length(robusta[861:903,2]);
```

```
Koers[1,3]<- var(EURUSD[8548:8562,5])/length(EURUSD[8548:8562,5]);  
Koers[2,3]<-var(EURUSD[8702:8706,5])/length(EURUSD[8702:8706,5]);  
Koers[3,3]<-var(EURUSD[8723:8729,5])/length(EURUSD[8723:8729,5]);  
Koers[4,3]<-var(EURUSD[9410:9416,5])/length(EURUSD[9410:9416,5]);
```

```
Koers[1,4]<- var(EURUSD[8530:8580,5])/length(EURUSD[8530:8580,5]);  
Koers[2,4]<-var(EURUSD[8686:8724,5])/length(EURUSD[8686:8724,5]);  
Koers[3,4]<-var(EURUSD[8705:8747,5])/length(EURUSD[8705:8747,5]);  
Koers[4,4]<-var(EURUSD[9392:9434,5])/length(EURUSD[9392:9434,5]);
```

```
Koers[1,5]<-var(VNDUSD[3534:3548,5])/length(VNDUSD[3534:3548,5]);  
Koers[2,5]<-var(VNDUSD[3688:3692,5])/length(VNDUSD[3688:3692,5]);  
Koers[3,5]<-var(VNDUSD[3709:3715,5])/length(VNDUSD[3709:3715,5]);  
Koers[4,5]<-var(VNDUSD[4396:4402,5])/length(VNDUSD[4396:4402,5]);
```

```
Koers[1,6]<-var(VNDUSD[3516:3564,5])/length(VNDUSD[3516:3564,5]);  
Koers[2,6]<-var(VNDUSD[3670:3710,5])/length(VNDUSD[3670:3710,5]);  
Koers[3,6]<-var(VNDUSD[3691:3733,5])/length(VNDUSD[3691:3733,5]);  
Koers[4,6]<-var(VNDUSD[4378:4420,5])/length(VNDUSD[4378:4420,5]);
```

```
Koers[1,7]<-var(BRLUSD[3952:3966,5])/length(BRLUSD[3952:3966,5]);  
Koers[2,7]<-var(BRLUSD[4106:4110,5])/length(BRLUSD[4106:4110,5]);
```

```
Koers [3,7] <- var(BRLUSD[4127:4133,5])/length(BRLUSD[4127:4133,5]);
Koers [4,7] <- var(BRLUSD[4814:4820,5])/length(EURUSD[4814:4820,5]);
```

```
Koers [1,8] <- var(BRLUSD[3934:3984,5])/length(BRLUSD[3934:3984,5]);
Koers [2,8] <- var(BRLUSD[3988:4128,5])/length(BRLUSD[3988:4128,5]);
Koers [3,8] <- var(BRLUSD[4109:4151,5])/length(BRLUSD[4109:4151,5]);
Koers [4,8] <- var(BRLUSD[4796:4838,5])/length(EURUSD[4796:4838,5]);
```

```
Verhoudingskoers <- matrix(NA,nrow=4,ncol=4)
```

```
Verhoudingskoers [1,1] <- Koers [1,1]/Koers [1,2];
Verhoudingskoers [2,1] <- Koers [2,1]/Koers [2,2];
Verhoudingskoers [3,1] <- Koers [3,1]/Koers [3,2];
Verhoudingskoers [4,1] <- Koers [4,1]/Koers [4,2];
```

```
Verhoudingskoers [1,2] <- Koers [1,3]/Koers [1,4];
Verhoudingskoers [2,2] <- Koers [2,3]/Koers [2,4];
Verhoudingskoers [3,2] <- Koers [3,3]/Koers [3,4];
Verhoudingskoers [4,2] <- Koers [4,3]/Koers [4,4];
```

```
Verhoudingskoers [1,3] <- Koers [1,5]/Koers [1,6];
Verhoudingskoers [2,3] <- Koers [2,5]/Koers [2,6];
Verhoudingskoers [3,3] <- Koers [3,5]/Koers [3,6];
Verhoudingskoers [4,3] <- Koers [4,5]/Koers [4,6];
```

```
Verhoudingskoers [1,4] <- Koers [1,7]/Koers [1,8];
Verhoudingskoers [2,4] <- Koers [2,7]/Koers [2,8];
Verhoudingskoers [3,4] <- Koers [3,7]/Koers [3,8];
Verhoudingskoers [4,4] <- Koers [4,7]/Koers [4,8];
```

```
Koers
```

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]
[1,]	1969.2330	634.1023	2.385795e-05	4.184104e-05	77.78095	190.48042
[2,]	1693.8600	1205.6436	3.426940e-05	1.127647e-04	489.96000	296.99244
[3,]	686.3333	114.4258	2.963041e-05	3.126262e-05	217.63946	472.93283
[4,]	1636.9932	540.2130	1.089068e-05	5.970283e-06	66.66667	32.65649

	[,7]	[,8]
[1,]	3.111319e-05	1.698920e-05
[2,]	3.906683e-03	3.307219e-04
[3,]	1.445042e-04	2.894568e-04
[4,]	2.113463e-05	6.165237e-06

Verhoudingskoers

	[,1]	[,2]	[,3]	[,4]
[1,]	3.105545	0.5702045	0.4083409	1.8313505
[2,]	1.404943	0.3039019	1.6497389	11.8125942
[3,]	5.998062	0.9477902	0.4601910	0.4992255
[4,]	3.030274	1.8241481	2.0414521	3.4280315

Verhoudingskoers [1,2]*Verhoudingskoers [1,3]*Verhoudingskoers [1,4]

[1] 0.4264077

Verhoudingskoers [2,2]*Verhoudingskoers [2,3]*Verhoudingskoers [2,4]

[1] 5.922349

Verhoudingskoers [3,2]*Verhoudingskoers [3,3]*Verhoudingskoers [3,4]

[1] 0.2177445

Verhoudingskoers [4,2]*Verhoudingskoers [4,3]*Verhoudingskoers [4,4]

[1] 12.76568

Koers.1<-matrix(NA,nrow=4,ncol=8)

Koers.1[1,1]<- var(robusta[80:90,2])/length(robusta[80:90,2]);

Koers.1[2,1]<-var(robusta[370:380,2])/length(robusta[370:380,2]);

Koers.1[3,1]<-var(robusta[580:600,2])/length(robusta[580:600,2]);

Koers.1[4,1]<-var(robusta[1100:1115,2])/length(robusta[1100:1115,2]);

Koers.1[1,2]<-var(robusta[62:108,2])/length(robusta[62:108,2]);

Koers.1[2,2]<-var(robusta[352:398,2])/length(robusta[352:398,2]);

Koers.1[3,2]<-var(robusta[562:618,2])/length(robusta[562:618,2]);

Koers.1[4,2]<-var(robusta[1082:1133,2])/length(robusta[1082:1133,2]);

Koers.1[1,3]<- var(EURUSD[8592:8602,5])/length(EURUSD[8592:8602,5]);

```
Koers.1[2,3]<-var(EURUSD[8882:8892,5])/length(EURUSD[8882:8892,5]);  
Koers.1[3,3]<-var(EURUSD[9092:9112,5])/length(EURUSD[9092:9112,5]);  
Koers.1[4,3]<-var(EURUSD[9612:9627,5])/length(EURUSD[9612:9627,5]);
```

```
Koers.1[1,4]<- var(EURUSD[8574:8620,5])/length(EURUSD[8574:8620,5]);  
Koers.1[2,4]<-var(EURUSD[8864:8910,5])/length(EURUSD[8864:8910,5]);  
Koers.1[3,4]<-var(EURUSD[9074:9130,5])/length(EURUSD[9074:9130,5]);  
Koers.1[4,4]<-var(EURUSD[9594:9645,5])/length(EURUSD[9594:9645,5]);
```

```
Koers.1[1,5]<-var(VNDUSD[3578:3588,5])/length(VNDUSD[3578:3588,5]);  
Koers.1[2,5]<-var(VNDUSD[3868:3878,5])/length(VNDUSD[3868:3878,5]);  
Koers.1[3,5]<-var(VNDUSD[4078:4098,5])/length(VNDUSD[4078:4098,5]);  
Koers.1[4,5]<-var(VNDUSD[4598:4613,5])/length(VNDUSD[4598:4613,5]);
```

```
Koers.1[1,6]<-var(VNDUSD[3560:3606,5])/length(VNDUSD[3560:3606,5]);  
Koers.1[2,6]<-var(VNDUSD[3850:3896,5])/length(VNDUSD[3868:3896,5]);  
Koers.1[3,6]<-var(VNDUSD[4060:4116,5])/length(VNDUSD[4060:4116,5]);  
Koers.1[4,6]<-var(VNDUSD[4580:4631,5])/length(VNDUSD[4580:4631,5]);
```

```
Koers.1[1,7]<-var(BRLUSD[3996:4006,5])/length(BRLUSD[3996:4006,5]);  
Koers.1[2,7]<-var(BRLUSD[4286:4296,5])/length(BRLUSD[4286:4296,5]);  
Koers.1[3,7]<-var(BRLUSD[4496:4516,5])/length(BRLUSD[4496:4516,5]);  
Koers.1[4,7]<-var(BRLUSD[5016:5031,5])/length(EURUSD[5016:5031,5]);
```

```
Koers.1[1,8]<- var(BRLUSD[3978:4028,5])/length(BRLUSD[3978:4028,5]);  
Koers.1[2,8]<-var(BRLUSD[4268:4314,5])/length(BRLUSD[4268:4314,5]);  
Koers.1[3,8]<-var(BRLUSD[4478:4534,5])/length(BRLUSD[4478:4535,5]);  
Koers.1[4,8]<-var(BRLUSD[4998:5049,5])/length(EURUSD[4998:5049,5]);
```

```
Verhoudingskoers.1 <- matrix(NA,nrow=4,ncol=4)
```

```
Verhoudingskoers.1[1,1]<-Koers.1[1,1]/Koers.1[1,2];  
Verhoudingskoers.1[2,1]<-Koers.1[2,1]/Koers.1[2,2];  
Verhoudingskoers.1[3,1]<-Koers.1[3,1]/Koers.1[3,2];  
Verhoudingskoers.1[4,1]<-Koers.1[4,1]/Koers.1[4,2];
```



```
Verhoudingskoers.1[1,2]<-Koers.1[1,3]/Koers.1[1,4];
Verhoudingskoers.1[2,2]<-Koers.1[2,3]/Koers.1[2,4];
Verhoudingskoers.1[3,2]<-Koers.1[3,3]/Koers.1[3,4];
Verhoudingskoers.1[4,2]<-Koers.1[4,3]/Koers.1[4,4];
```

```
Verhoudingskoers.1[1,3]<-Koers.1[1,5]/Koers.1[1,6];
Verhoudingskoers.1[2,3]<-Koers.1[2,5]/Koers.1[2,6];
Verhoudingskoers.1[3,3]<-Koers.1[3,5]/Koers.1[3,6];
Verhoudingskoers.1[4,3]<-Koers.1[4,5]/Koers.1[4,6];
```

```
Verhoudingskoers.1[1,4]<-Koers.1[1,7]/Koers.1[1,8];
Verhoudingskoers.1[2,4]<-Koers.1[2,7]/Koers.1[2,8];
Verhoudingskoers.1[3,4]<-Koers.1[3,7]/Koers.1[3,8];
Verhoudingskoers.1[4,4]<-Koers.1[4,7]/Koers.1[4,8];
```

```
par(mfrow=c(2,2))
ts.plot(robusta[80:90,2])
ts.plot(robusta[370:380,2])
ts.plot(robusta[580:600,2])
ts.plot(robusta[1100:1115,2])
```

```
Verhoudingskoers.1[1,2]*Verhoudingskoers.1[1,3]*Verhoudingskoers.1[1,4]
[1] 0.1621877
Verhoudingskoers.1[2,2]*Verhoudingskoers.1[2,3]*Verhoudingskoers.1[2,4]
[1] 12.17629
Verhoudingskoers.1[3,2]*Verhoudingskoers.1[3,3]*Verhoudingskoers.1[3,4]
[1] 0.02725991
Verhoudingskoers.1[4,2]*Verhoudingskoers.1[4,3]*Verhoudingskoers.1[4,4]
[1] 0.5918086
```

```
par(mfrow=c(2,2))
ts.plot(robusta[36:50,2])
ts.plot(robusta[187:191,2])
ts.plot(robusta[208:214,2])
ts.plot(robusta[879:885,2])
```

```
Verhoudingskoers [1,2]*Verhoudingskoers [1,3]*Verhoudingskoers [1,4]
[1] 0.4264077
```

```
Verhoudingskoers [2,2]*Verhoudingskoers [2,3]*Verhoudingskoers [2,4]
[1] 5.922349
```

```
Verhoudingskoers [3,2]*Verhoudingskoers [3,3]*Verhoudingskoers [3,4]
[1] 0.2177445
```

```
Verhoudingskoers [4,2]*Verhoudingskoers [4,3]*Verhoudingskoers [4,4]
[1] 12.76568
```

```
diff((robusta[36:50,2]),lag=1)
```

```
[1] 3 1 29 -163 -119 12 36 70 89 -110 -185 10 -111 -92
```

```
diff((robusta[187:191,2]),lag=1)
```

```
[1] -24 -105 10 -111
```

```
diff((robusta[208:214,2]),lag=1)
```

```
[1] 3 114 20 -74 4 120
```

```
diff((robusta[879:885,2]),lag=1)
```

```
[1] -44 -36 104 59 114 -16
```

.1.17 Script 17

Hieronder staat het script waarmee de vorm van de tijdsreeksen is bekeken.

```
plot(density(robusta$X2nd))
```

```
plot(density(diff(robusta$X2nd,lag=1)))
```

```
x<-rnorm(length(diff(robusta$X2nd,lag=1)),mean=mean(diff(robusta$X2nd,lag=1)),sd=sd
```

```
plot(density(x))
```

```
y.1<-rt(length(diff(robusta$X2nd,lag=1)),df=1)
```

```
plot(density(diff(robusta$X2nd,lag=1)))
```

```
length(diff(robusta$X2nd,lag=1));mean(diff(robusta$X2nd,lag=1));sd(diff(robusta$X2nd,
```

```
min(diff(robusta$X2nd,lag=1));range(diff(robusta$X2nd,lag=1));skewness(diff(robusta$
```

```
[1] 1268
```

```
[1] -0.03706625
```

```
[1] 34.97255
```

```
[1] 159
```

```
[1] -185
```

```
[1] -185 159  
[1] -0.206767  
[1] 2.442099
```

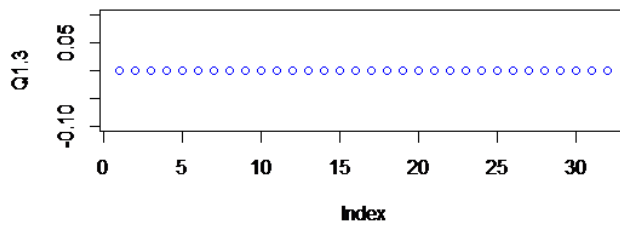
```
s<-sort(diff(robusta$X2nd,lag=1))  
s1<-matrix(NA,nrow=634,ncol=2)  
for (i in 1:634){s1[i,1]<-s[635-i];s1[i,2]<-s[634+i];}  
plot(s1[,1],s1[,2])  
abline(a=0,b=-1,col="red",lty=2)
```

```
U<-matrix(NA,nrow=1268)  
for (i in 1:1268){ U[i,1]=(diff(robusta$X2nd,lag=1)[i]-mean(diff(robusta$X2nd,lag=1),  
mean(U)  
[1] -9.710792e-18  
var(U)  
[,1]  
[1,] 1  
ts.plot(U,main="Plot van U_t behorende bij X_t=mu+U_t*sigma_t")
```

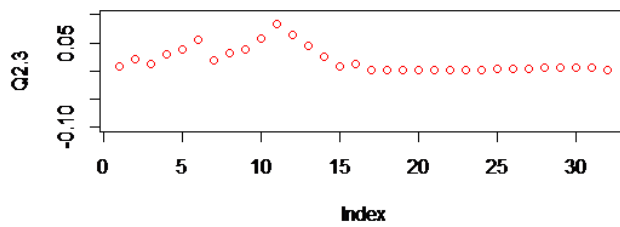
```
plot(density(U))
```

.2 Figuren

p-waardes behorende bij de ACF van de robusta-set, 3 maanden

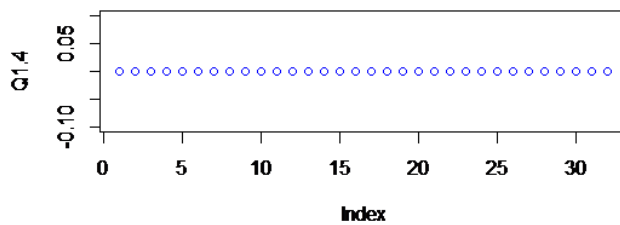


irides behorende bij de ACF van de robusta-set, 3 maanden, met lag=1

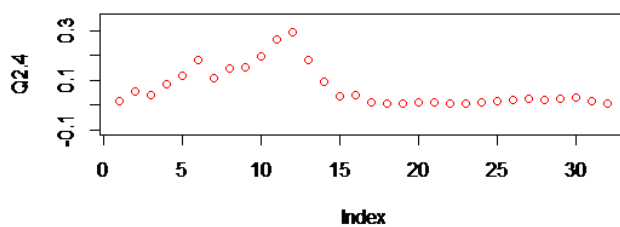


Figuur 8: Plot van de p waardes van de robusta set, 3 maanden, bovenste zonder lag, tweede met lag=1, 2e kolom

p-waardes behorende bij de ACF van de robusta-set, 4 maanden

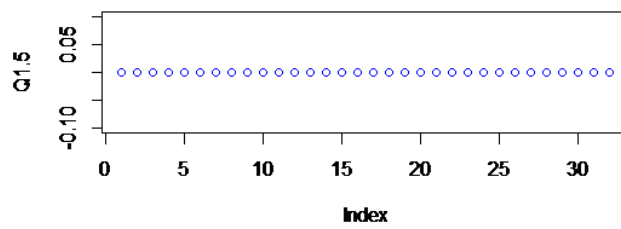


irides behorende bij de ACF van de robusta-set, 4 maanden, met lag=1

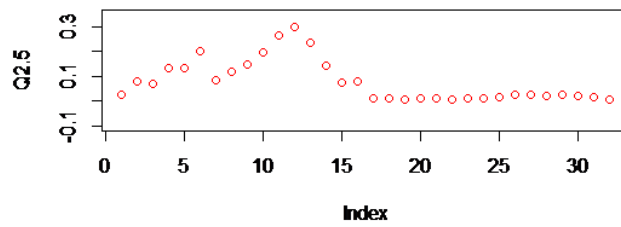


Figuur 9: Plot van de p waardes van de robusta set, 4 maanden, bovenste zonder lag, tweede met lag=1, 2e kolom

p-waardes behorende bij de ACF van de robusta-set, 5 maanden



irides behorende bij de ACF van de robusta-set, 5 maanden, met lag=1



Figuur 10: Plot van de p waardes van de robusta set, 5 maanden, bovenste zonder lag, tweede met lag=1, 2e kolom

Bibliografie

- [1] *Autoregressive conditional heteroskedasticity*, September 2013. http://en.wikipedia.org/wiki/Autoregressive_conditional_heteroskedasticity.
- [2] *Brownse beweging (wiskunde)*, Maart 2013. [http://nl.wikipedia.org/wiki/Brownse_beweging_\(wiskunde\)](http://nl.wikipedia.org/wiki/Brownse_beweging_(wiskunde)).
- [3] *Ito's Lemma*, Augustus 2013. http://en.wikipedia.org/wiki/It%C5%8D's_lemma.
- [4] *Kurtosis*, Maart 2013. <http://nl.wikipedia.org/wiki/Kurtosis>.
- [5] Ali, Ghulam: *EGARCH, GJR-GARCH, TGARCH, AVGARCH, NGARCH, IGARCH and APARCH Models for Pathogens at Marine Recreational Sites*. *Journal of Statistical and Econometric Methods*, 2(3):57-73, 2013. http://www.scienpress.com/Upload/JSEM/Vol%202_3_6.pdf.
- [6] Hafner, Jrgen Franke Wolfgang Hrdle Christian: *Introduction to Statistics of Financial Markets*, volume 424. Springer, 1st edition, 1st uitgave, April 2004. http://sfb649.wiwi.hu-berlin.de/fedc_homepage/explore/ebooks/html/sfe/sfenode68.html, Paragraaf 13.3.
- [7] Hagerud, Gustaf: *A New Non-Linear GARCH Model*. Gotab, Handelshogskolan, Sveavagen 65, 1997. <http://www.diva-portal.org/smash/get/diva2:221859/FULLTEXT01.pdf>.
- [8] Mills, Terence C.: *The Econometric Modelling of Financial Time Ser.* Press Syndicate of the University of Cambridge, 1999. Hoofdstuk3.
- [9] Mills, Terence C.: *The Econometric Modelling of Financial Time Series.* Press Syndicate of the University of Cambridge, 1999. Hoofdstuk2.

- [10] Mills, Terence C.: *The Econometric Modelling of Financial Time Series*. Press Syndicate of the University of Cambridge, 1999. Hoofdstuk4.
- [11] Song, Xiaojun: *Multivariate GARCH Models for the Greater China Stoch Markets*. Dissertations and Theses, 64:14, 2009. http://ink.library.smu.edu.sg/cgi/viewcontent.cgi?article=1029&context=etd_coll.