Robust time series analysis with R

Alexander Dürre

ECARES - European Center for Advanced Research in Economics and Statistics
Université Libre de Bruxelles

Wien, 20th September 2021

Exercises A

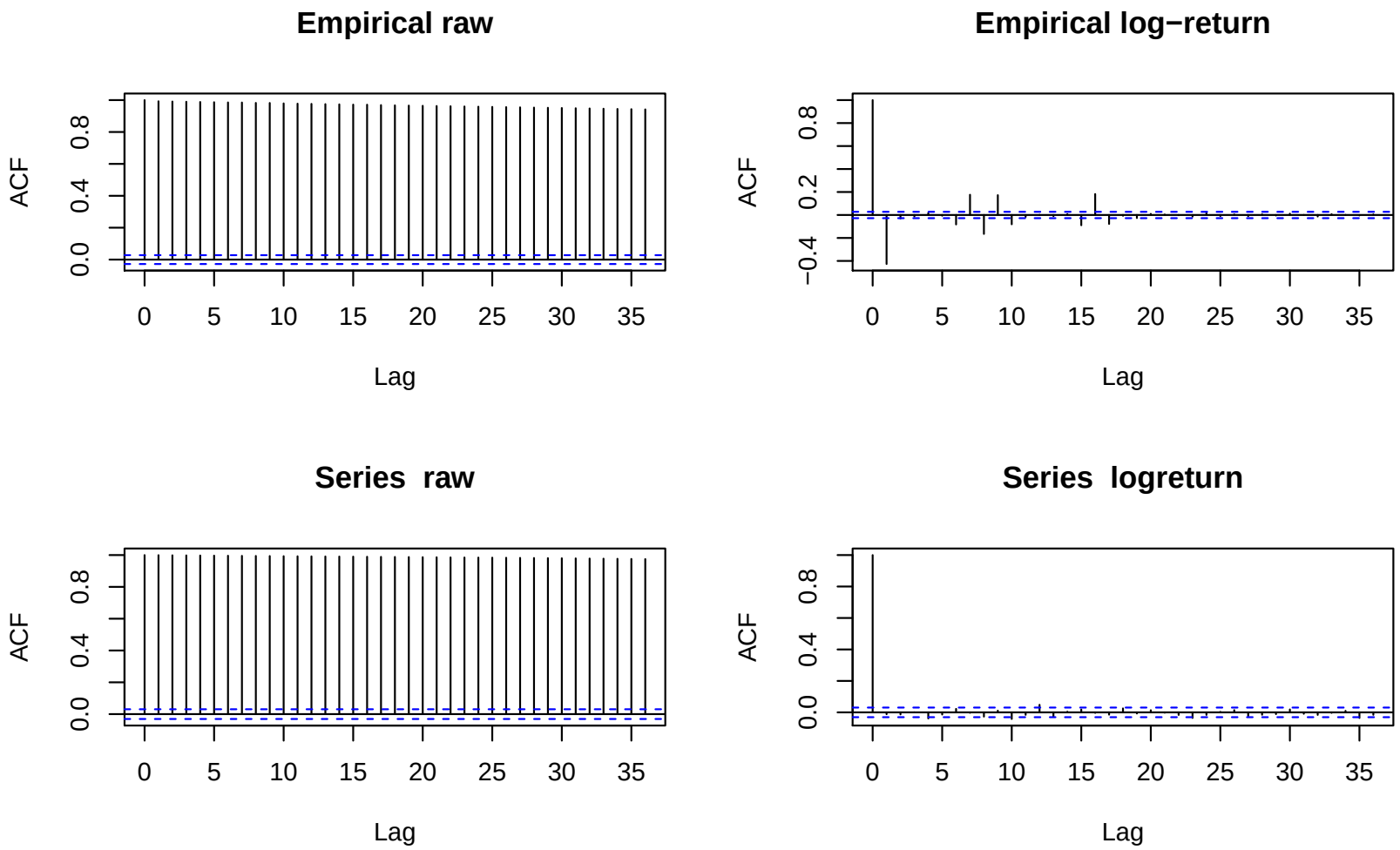
1. Calculate the ACF of the stock prices of SAP (SAP2.Rdata) robustly and by the empirical ACF. consider the raw prices and the log returns defined by $(\tilde{X}_t)_{t=2,\ldots,n}$ defined by $\tilde{X}_t = \log(X_t) - \log(X_{t-1}))$.
2. Determine the frequency of the guitar string (guitar2.Rdata)! (Which string is played and is it tuned correctly?)
3. Smooth the NO2 values robustly and non robustly. Decide for a reasonable period length (and specify it by the freq argument in the ts-object)

Note: install `robts` (after installing dependencies: robustbase, rrcov, SpatialNP, ICSNP, sscor, quantreg, ltsa) by:

```
install.packages("robts", repos="http://R-Forge.R-project.org")
```

A1

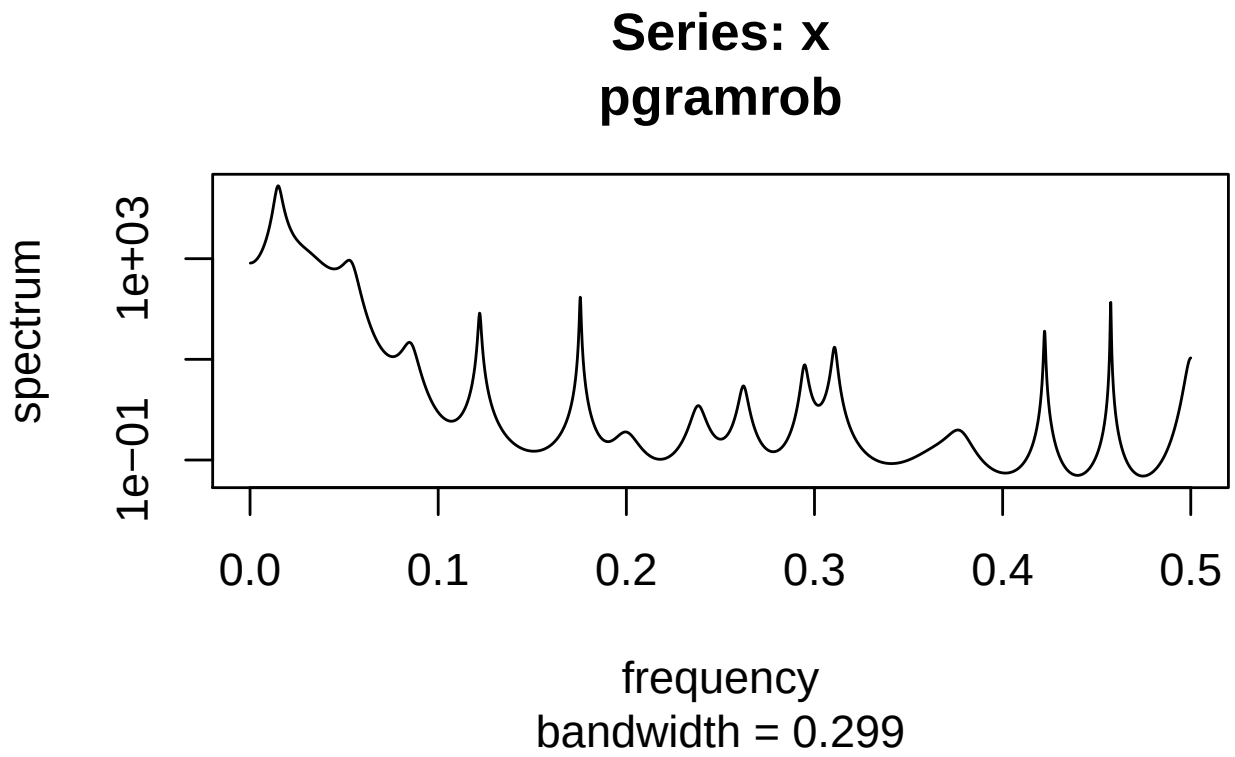
```
> load("SAP2.Rdata")
> raw <- SAP[,2]
> logreturn <- diff(log(SAP[,2]))
> acf(raw)
> acf(logreturn)
> acfrob(raw)
> acfrob(logreturn)
```



Empirical ACFs in the forst row, robust ACFs in the second row

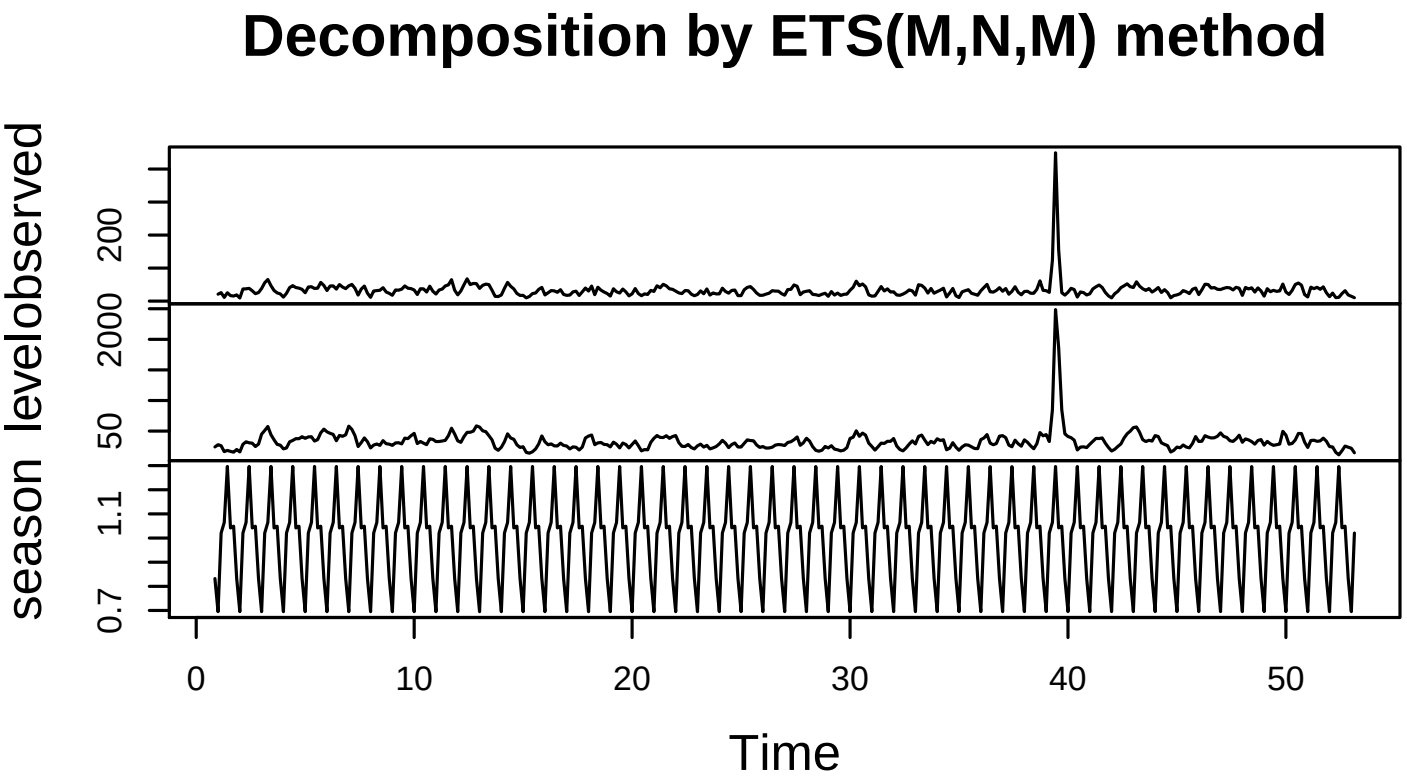
```
> spectrumg <- spectrumrob(guitar[,2])
> index <- which.max(spectrumg$spec)
> spectrumg$freq[index]*8000
```

the frequency is 119Hz which is a little to high for an *A*string

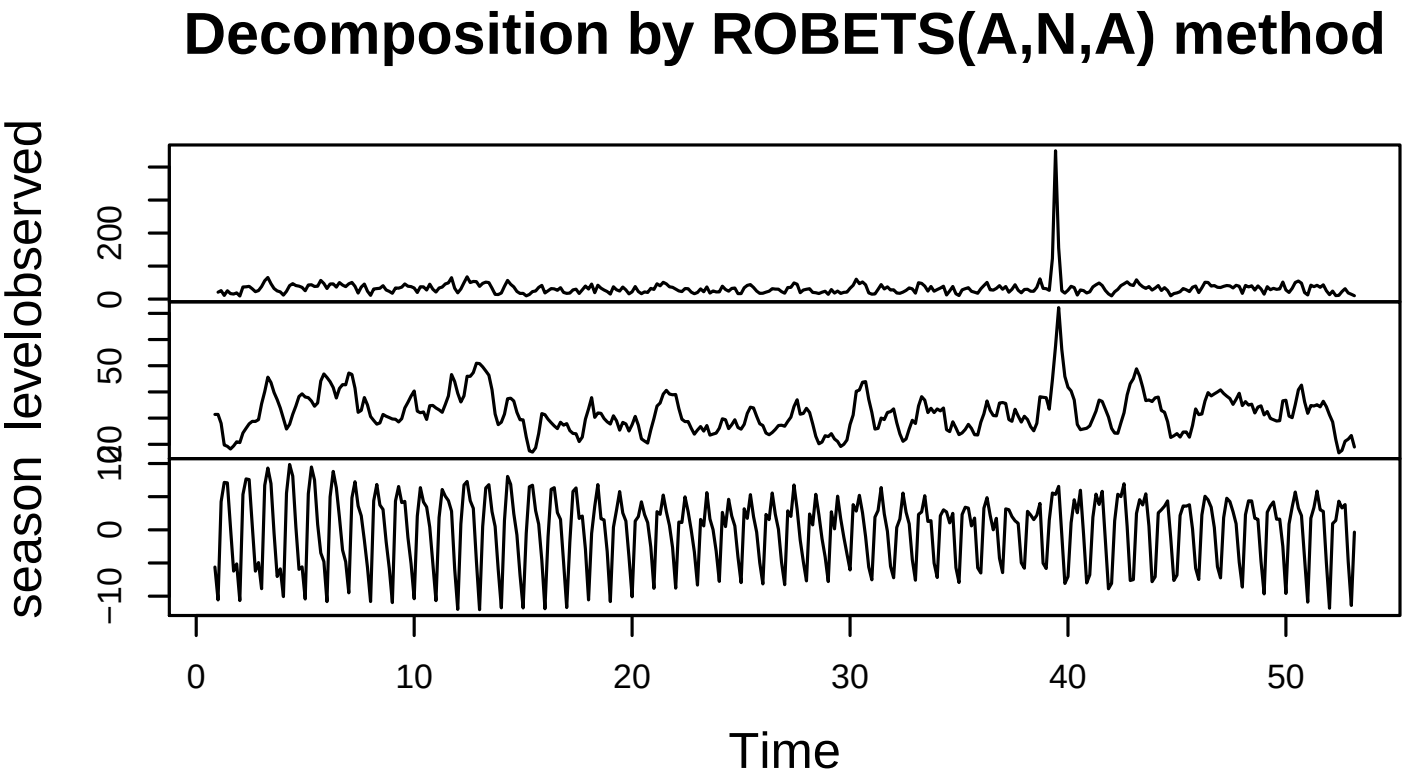


A3

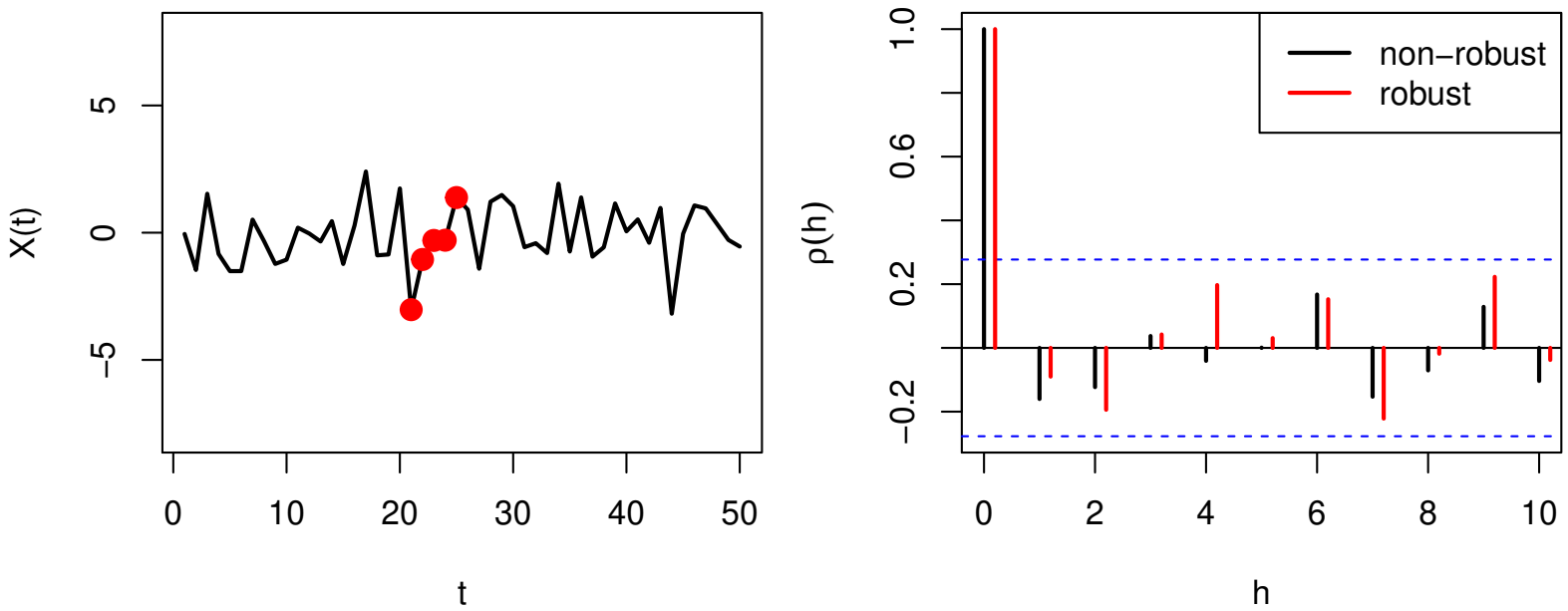
```
>load("NO2Krefeld.Rdata")
>KrefeldNO2 <- ts(Krefeld[,2],freq=7)
>etsresult  <- ets(KrefeldNO2)
>robetsresult <- robets(KrefeldNO2)
>plot(etsresult)
>plot(robetsresult)
```



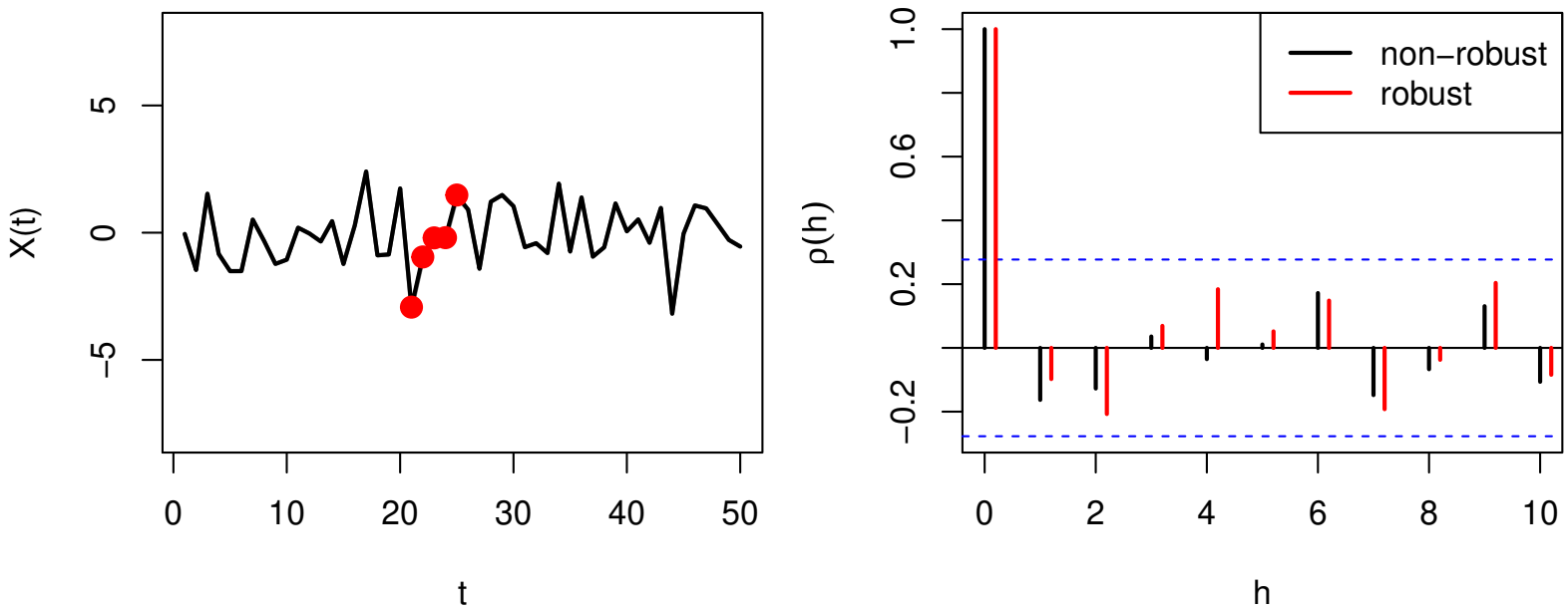
Non-robust smoothing



Robust smoothing



AR(1) with $\pi = 0.9$ and one outlier (left) and estimated ACF (right)



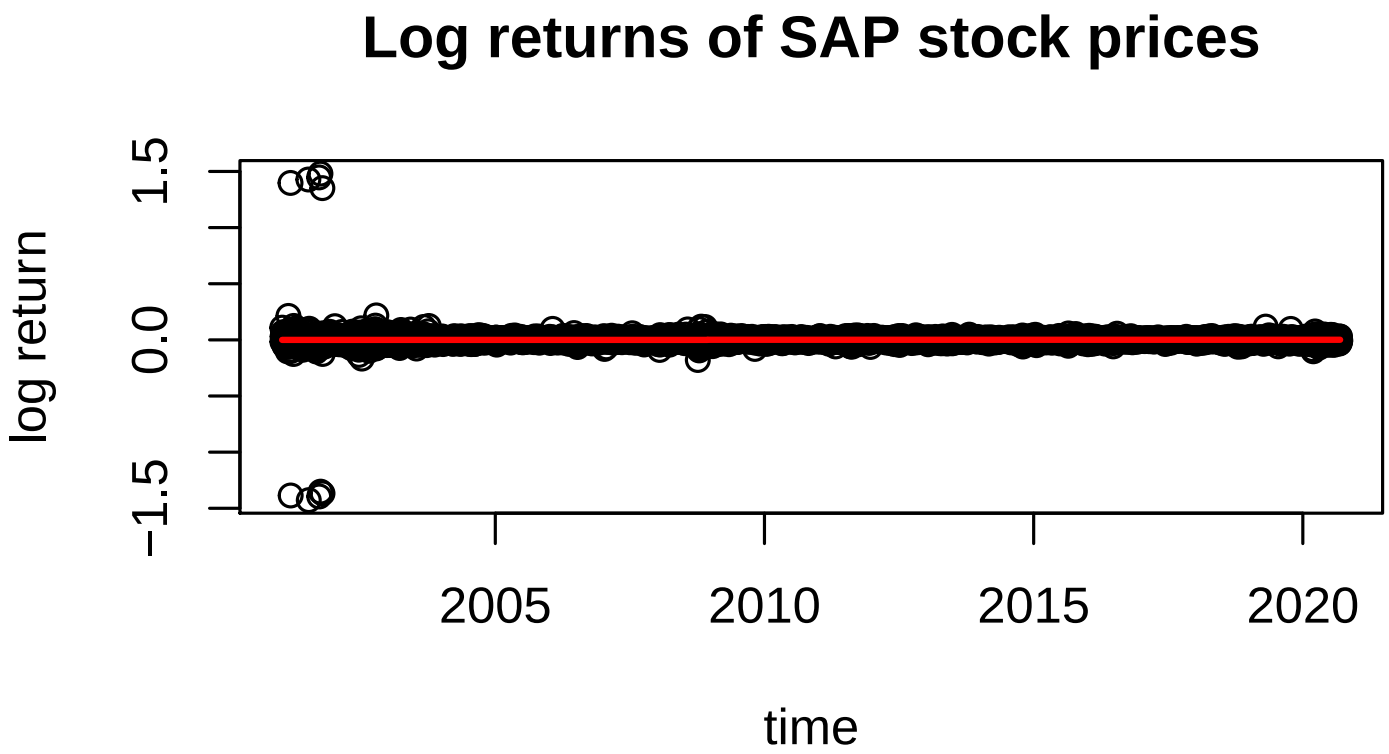
AR(1) with $\pi = 0.9$ and one outlier (left) and estimated ACF (right)

Exercises

1. Detect changes in the location and scale of the log returns of the SAP stock prices. If you can detect a change (significance level 0.05), spit the time series in two and try to detect changes there.
2. Fit an ARMA model to the yearly sunhours of Chemnitz robustly, try also conventional fits of order (1,0),(0,1) an (1,1).

```
> logreturns <- diff(log(SAP[,2]))
> test1 <- huber_cusum(logreturns,fun="HLm")
> test1$p.value
> test1$cp.location
> test2a <- huber_cusum(logreturns[1:2008],fun="HLm")
> test2b <- huber_cusum(logreturns[2009:4984],fun="HLm")
> test2a$p.value
> test2b$p.value
> median(logreturns[1:2008])-median(logreturns[2009:4984])
```

difference is around 0.001!



Estimated location function

Exercises

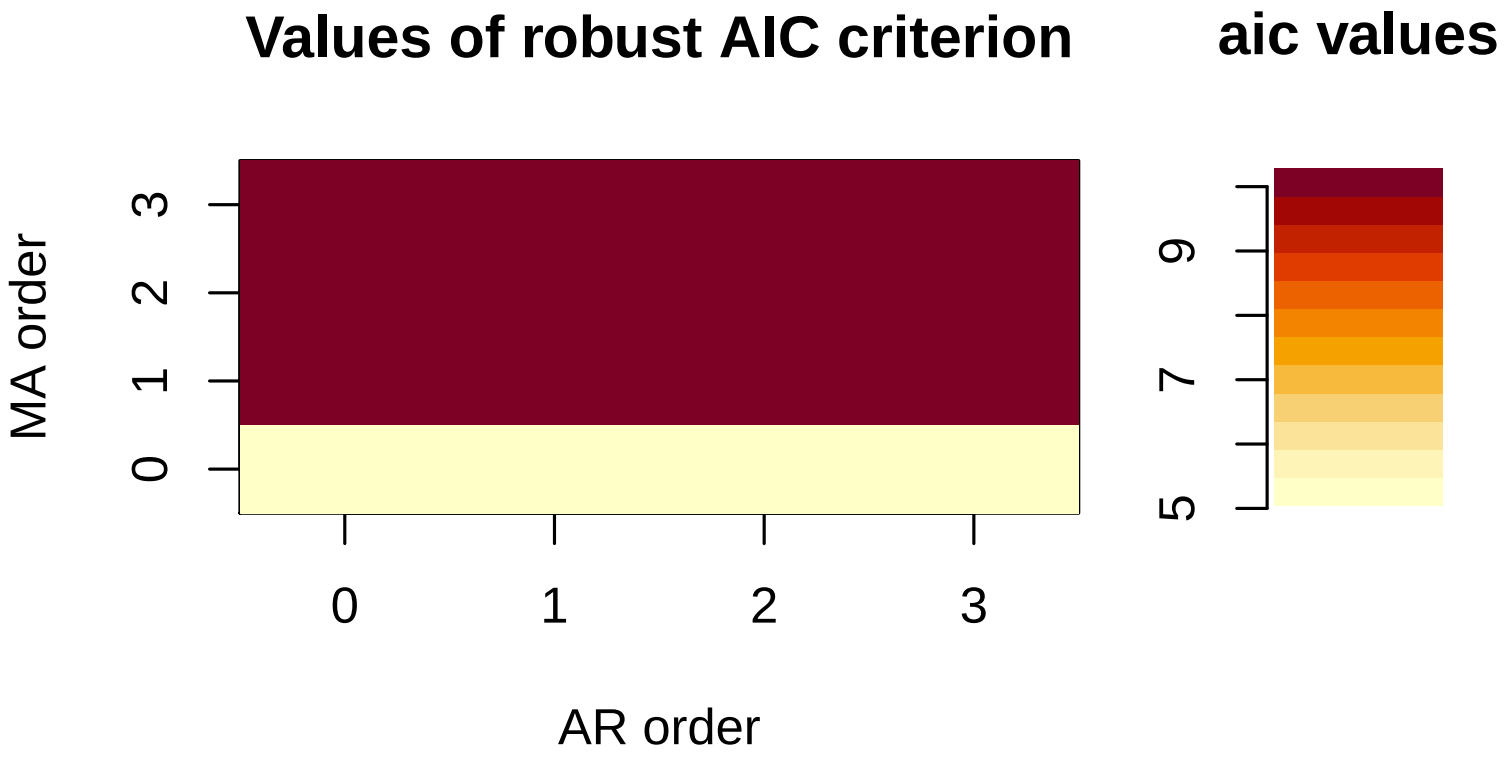
1. Try to predict the number of covid cases in Austria for the next days.

```
> armarobM <- armarob(sunhours[,2],arorder=3,maorder=3,
  aic=TRUE,aicpenalty=function(p) return(p*log(length(sunho
> arma10 <- arma(sunhours[,2],c(1,0))
> arma01 <- arma(sunhours[,2],c(0,1))
> arma11 <- arma(sunhours[,2],c(1,1))
```

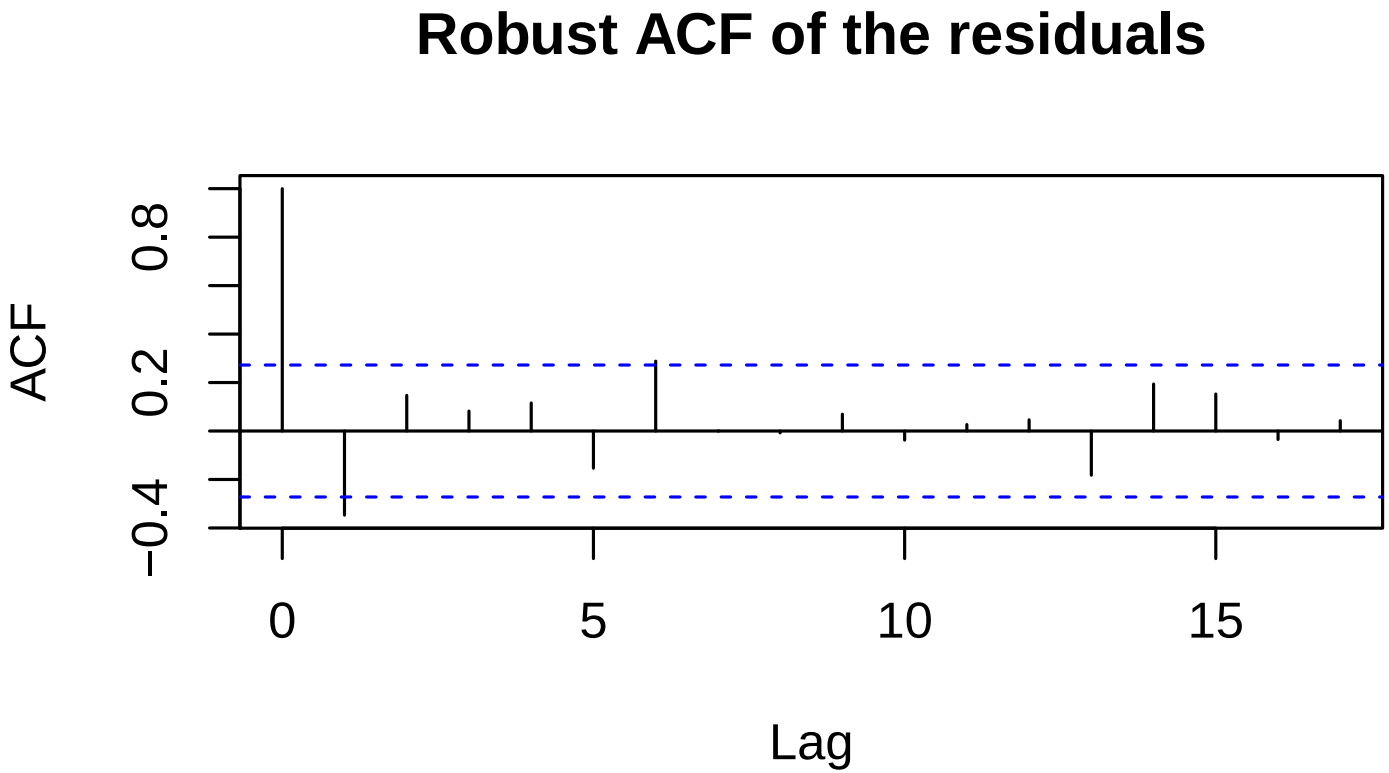
only robust ARMA model [AR(1)] and MA(1) are reasonable

model	μ	π	θ
robust	1603	0.51	0
non robust AR(1)	1197	0.26	
non robust ARMA(1,1)	1196	0.78	-0.58
non robust MA(1)	1611	0	0.23

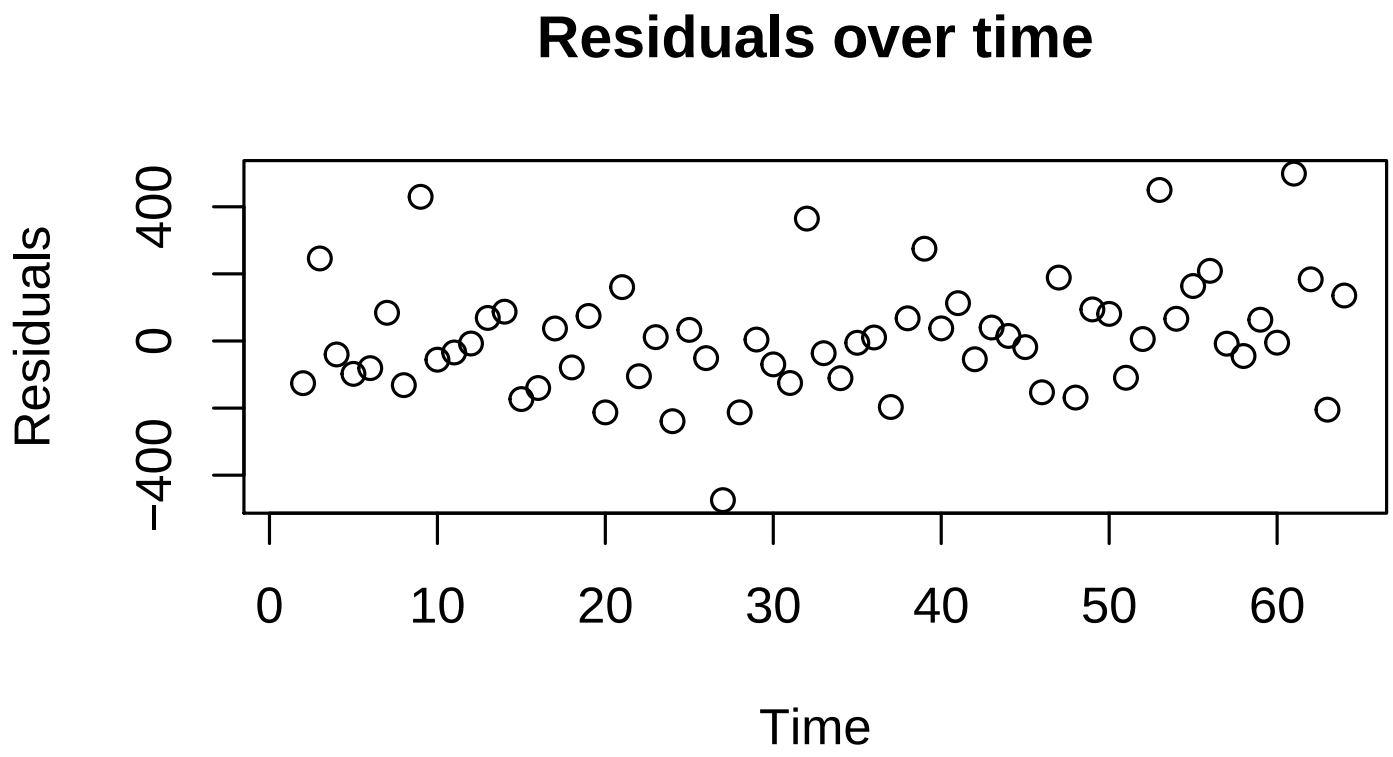
Parameter estimations



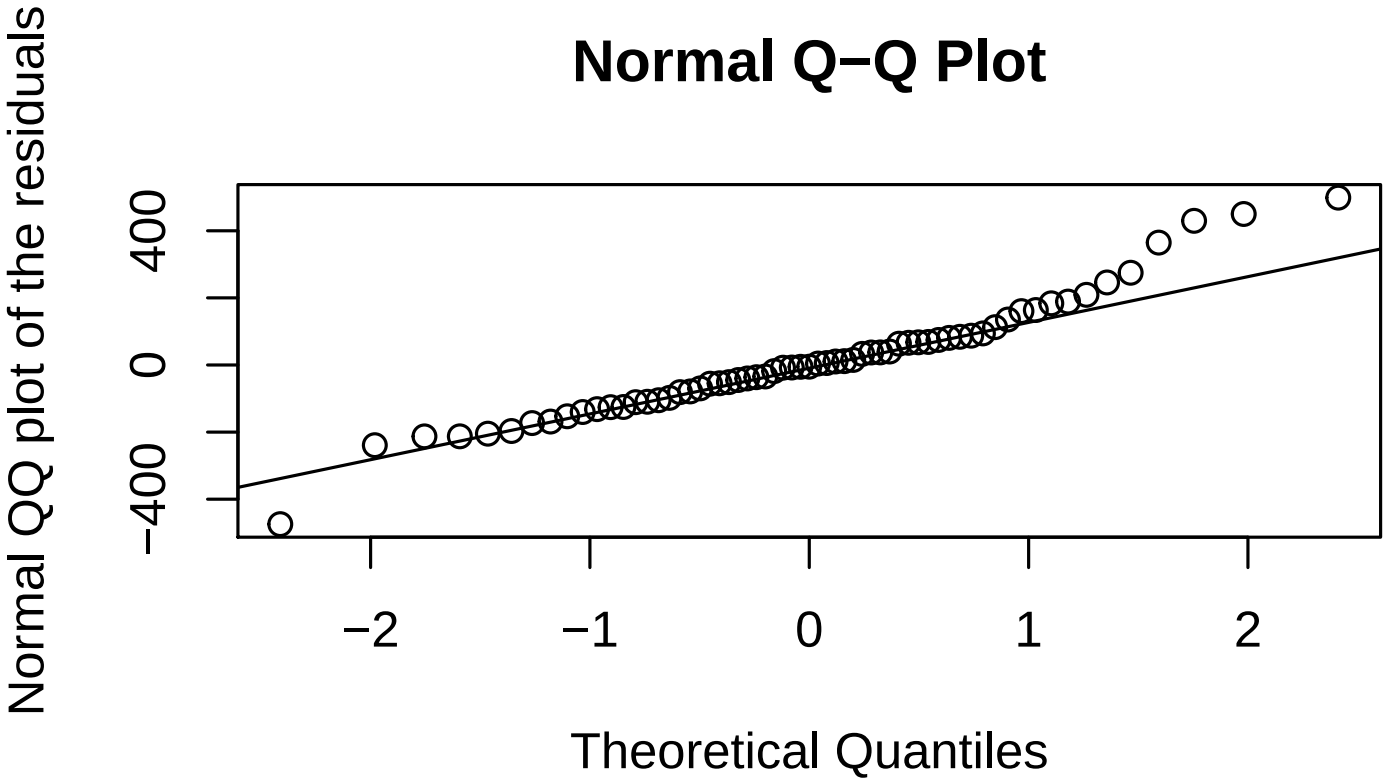
Model analysis



Model analysis

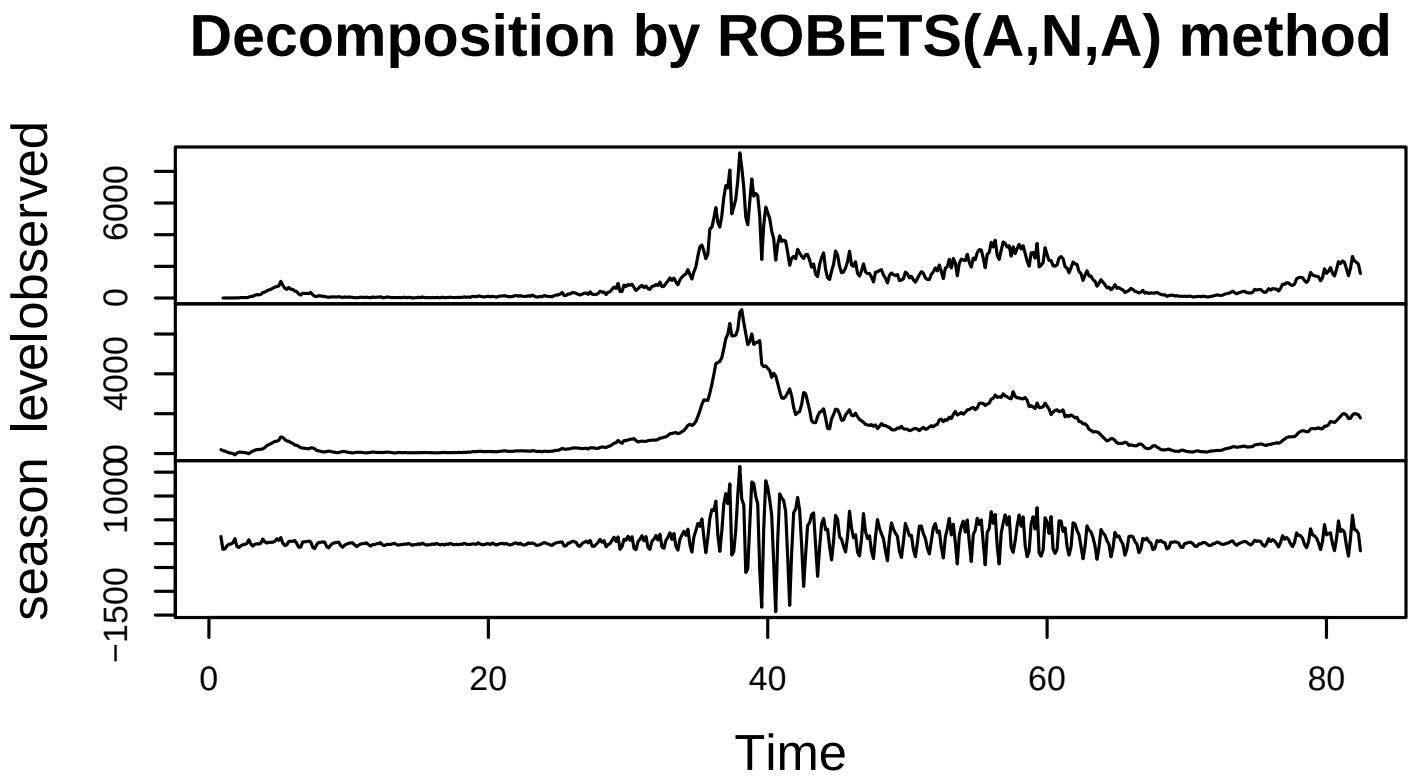


Model analysis



Model analysis

```
> load("Covid.Rdata")
> covidtimes <- ts(Covid[,5],freq=7)
> robModel <- robets(covidtimes)
> predict(robModel)
```



Robustly smoothed covid data

Data sources:

- `https://open.data.dwd.de`
- `https://finance.yahoo.com`
- `https://covid19-dashboard.ages.at`
- `https://www.lanuv.nrw.de`