Solutions\_Ramon\_Schildknecht\_RTP Excercise Sheet - Series 2

library(tseries)  
library(tidyverse)  
library(magrittr)  
library(plotly)  
library(timetk)  
library(lubridate)  
library(pracma)  
library(fma)

# Template

# to do

# to do

# to do

# to do

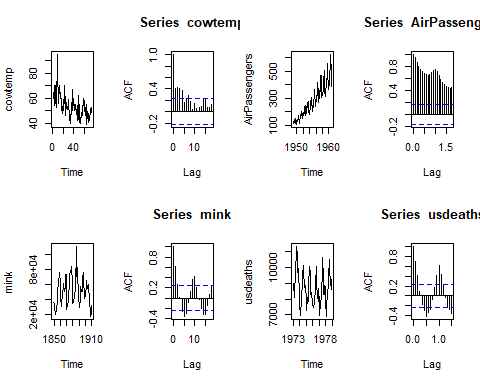
# Series 2

## Exercise 2.1 - Match plots to correlograms

Assuming the following solutions by comparing the given pictures: A -> 4, B -> 3, C -> 2, D -> 1

Checking by code:

par(mfrow=c(2,4))  
# Data set cow temperature  
plot(cowtemp)  
acf(cowtemp)  
# Data set air passengers  
plot(AirPassengers)  
acf(AirPassengers)  
# Data set mink trappings  
plot(mink)  
acf(mink)  
# Data set accidential deaths in the US  
plot(usdeaths)  
acf(usdeaths)



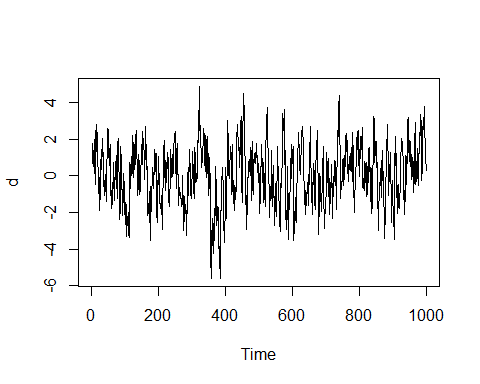
True results: A -> 2, B -> 3, C -> 1, D -> 4

## Exercise 2.2 - calculate the lagged scatter plot and the plug-in estimator

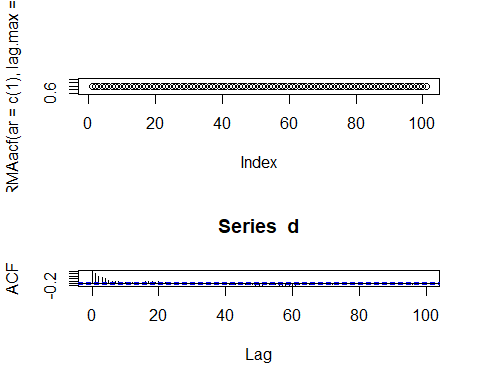
skip for the moment

## Exercise 2.3 - AR(1) process

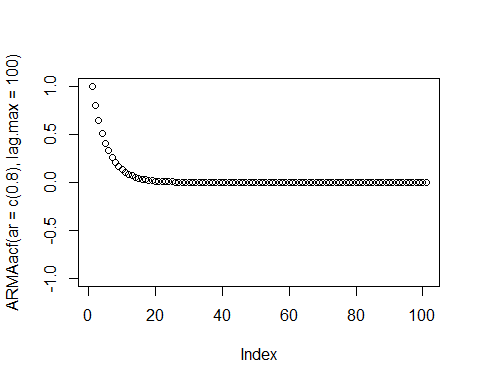
d <- arima.sim(list(ar = c(0.8)), n = 1000)  
plot(d)



par(mfrow=c(2,1))  
plot(ARMAacf(ar = c(1), lag.max = 100))  
acf(d, lag.max = 100)

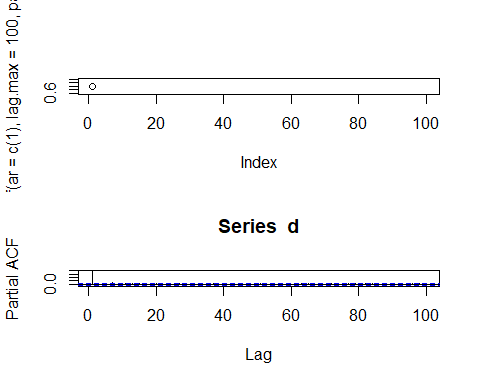


# assumption k = 100 as before  
plot(ARMAacf(ar = c(0.8), lag.max = 100), ylim = c(-1, 1))



Dependence is decreasing fast from Lag 1 to lag 20. Afterwards the dependence continues stable towards zero.

par(mfrow=c(2,1))  
plot(ARMAacf(ar = c(1), lag.max = 100, pacf = TRUE))  
pacf(d, lag.max = 100)



Observation: good agreement only for the first lag. Theoretical PACF just generates a value for lag 1. All other values are NAs!

## Exercise 2.4

d <- read\_table2("kreatin.txt")

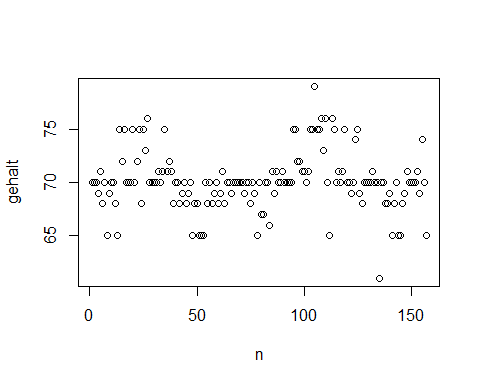
## Warning: Missing column names filled in: 'X3' [3]

## Parsed with column specification:  
## cols(  
## n = col\_integer(),  
## gehalt = col\_integer(),  
## X3 = col\_character()  
## )

## Warning in rbind(names(probs), probs\_f): number of columns of result is not  
## a multiple of vector length (arg 1)

## Warning: 157 parsing failures.  
## row # A tibble: 5 x 5 col row col expected actual file expected <int> <chr> <chr> <chr> <chr> actual 1 1 <NA> 3 columns 2 columns 'kreatin.txt' file 2 2 <NA> 3 columns 2 columns 'kreatin.txt' row 3 3 <NA> 3 columns 2 columns 'kreatin.txt' col 4 4 <NA> 3 columns 2 columns 'kreatin.txt' expected 5 5 <NA> 3 columns 2 columns 'kreatin.txt'  
## ... ................. ... ............................................... ........ ............................................... ...... ............................................... .... ............................................... ... ............................................... ... ............................................... ........ ...............................................  
## See problems(...) for more details.

d %<>% select(-X3)  
plot(d)



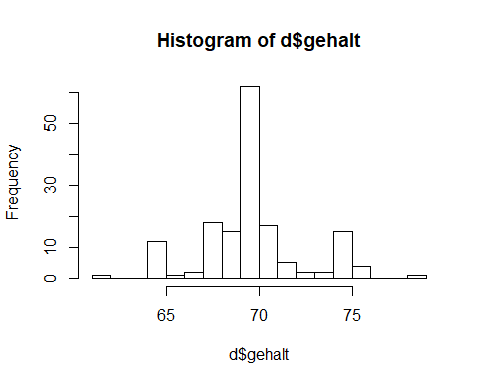
summary(d$gehalt)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 61.00 69.00 70.00 70.12 71.00 79.00

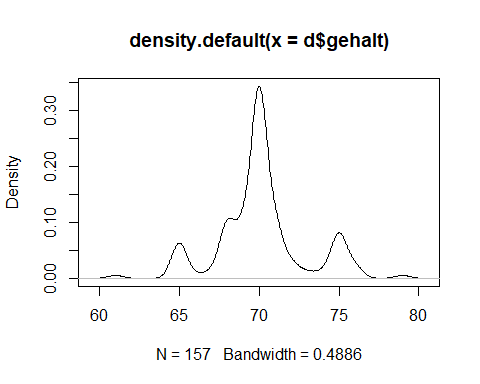
sd(d$gehalt)

## [1] 2.758089

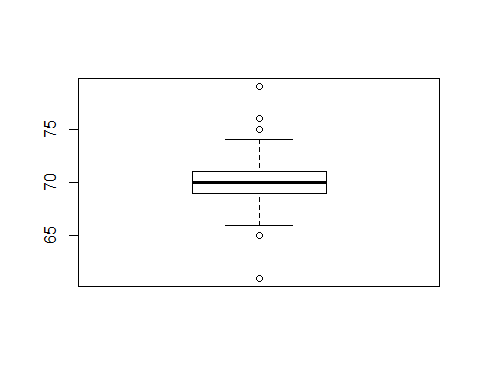
hist(d$gehalt, breaks = 25)



plot(density(d$gehalt))

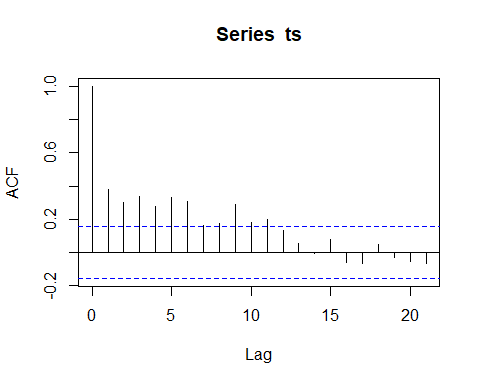


boxplot(d$gehalt)

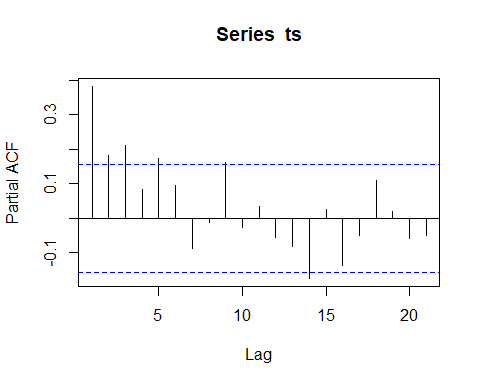


Assumption: There should be constanct values with error. Meaning: It should be a stationary process. Generated plots above do not confirm the assumption completly. The following aspect is not given: All Xt are not identically distributed.

ts <- ts(d$gehalt)  
acf(ts)



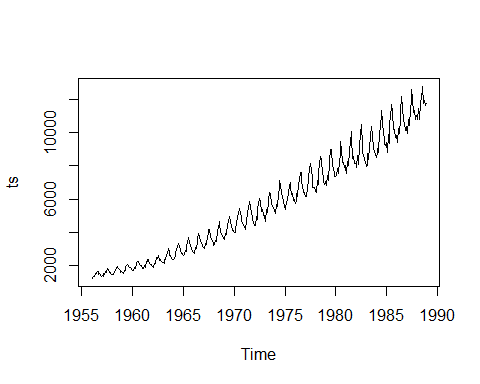
pacf(ts)



Insights do not confirm a stationary process. There are correlations at multiple lags (e. g. 2, 3, 4, 9). One could recognize partial autocorrelations at multiple lags as well. Conclusion: One could recognize that autocovariance dependends not only at lag h. Next possible step: Use differencing at lag 1 and check acf as well as pacf again.

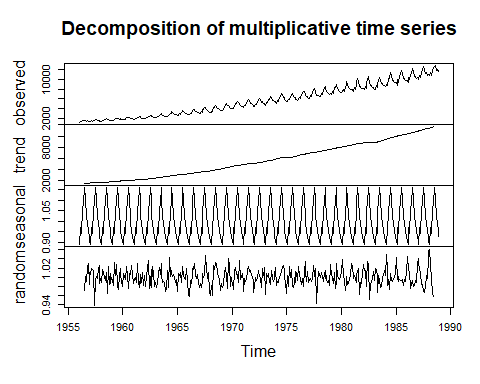
## Exercise 2.5

d <- read.table("cbe.dat.txt")  
ts <- ts(d, frequency = 12, start = c(1956,1))  
ts <- ts[, "elec"]  
plot(ts)

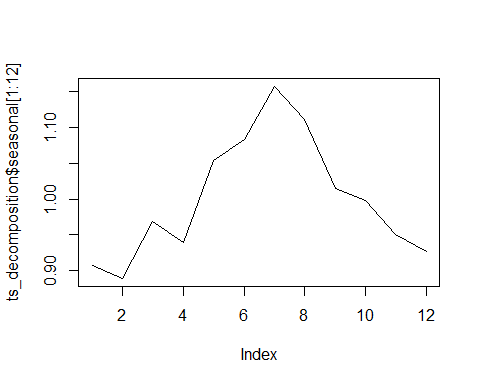


Because one can see a clear trend as well as a seasonality. We need to adress this issues first.

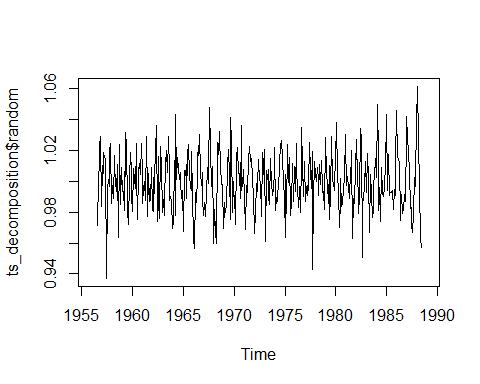
plot(ts\_decomposition <- decompose(ts, type = "multiplicative"))



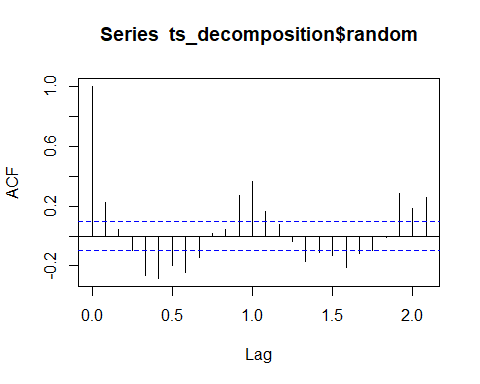
plot(ts\_decomposition$seasonal[1:12], type = "l") # one saison



plot(ts\_decomposition$random)



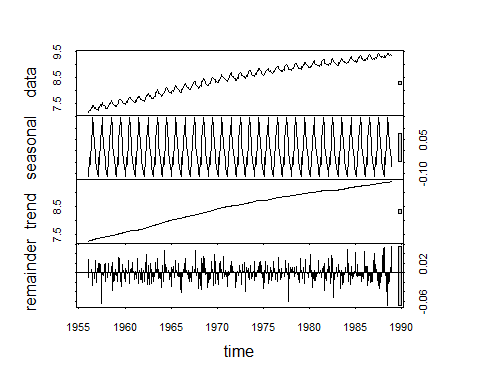
acf(ts\_decomposition$random, na.action = na.pass, plot = TRUE)



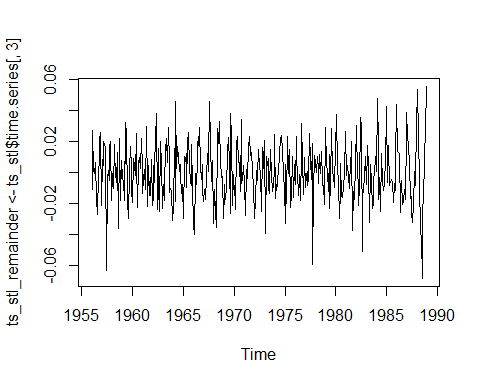
One sees a clear increasing trend as well as a yearly saisonality with peaks in summer. The remainder represents structure, e. g. at lag 9 to 11. Conclusion: The residuals don’t seem to appear random. That is an issue to be solved.

Averaging window approach

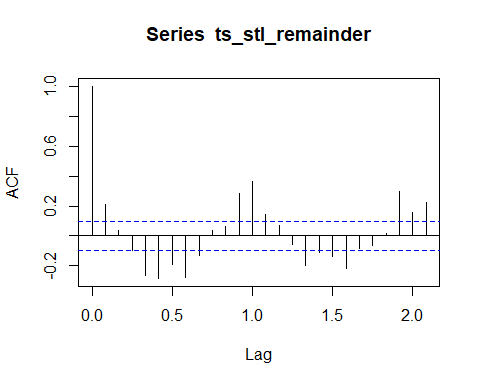
plot(ts\_stl <- stl(log(ts), s.window = "periodic"))



plot(ts\_stl\_remainder <- ts\_stl$time.series[,3])



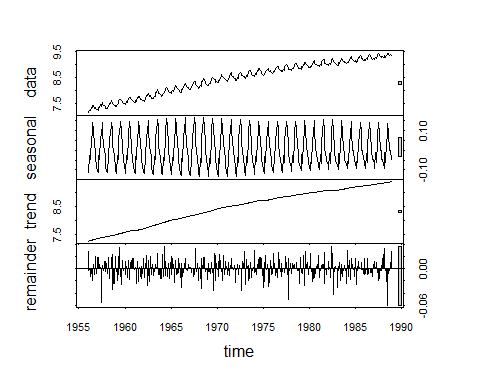
acf(ts\_stl\_remainder)



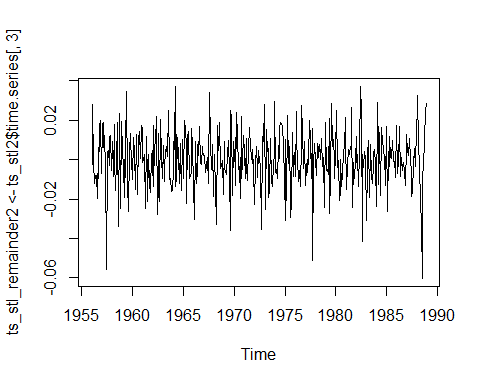
Same as above in b): One sees a clear increasing trend as well as a yearly saisonality with peaks in summer. The remainder represents structure, e. g. at lag 9 to 11. Conclusion: The residuals don’t seem to appear random. That is an issue to be solved.

Smoothing window approach

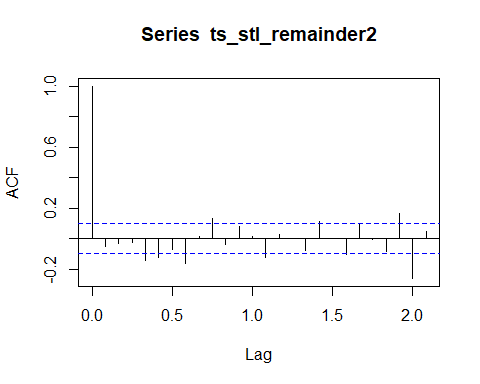
plot(ts\_stl2 <- stl(log(ts), s.window = 12)) # assumption: 12 means a year span in this case



plot(ts\_stl\_remainder2 <- ts\_stl2$time.series[,3])



acf(ts\_stl\_remainder2)

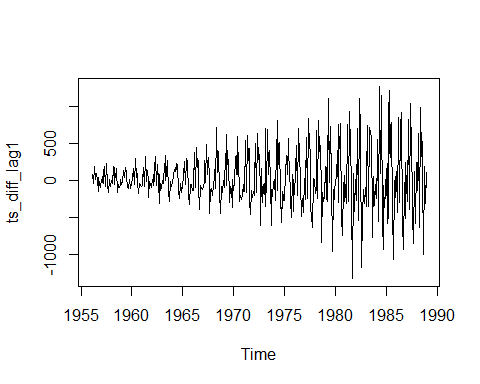


Smooting window approach worked better than the two previous approaches. The remainder seems pretty stochastic with little signs of autocorrelation at lag 8, 19 and 20. Hint: we didn’t cover residual analyis so far. Therefore we do not dive deeper regarding this topic now.

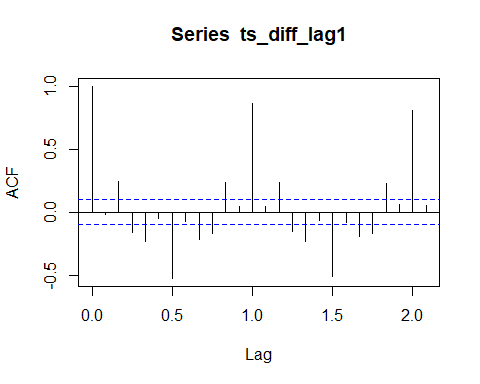
Parameter type = “multiplicative”: Because there is a seasonality and a trend at the same time and both are changing over time (e. g. their variation). Log-transformation: Needed to transform the multiplicative to an additive model.

lag 1

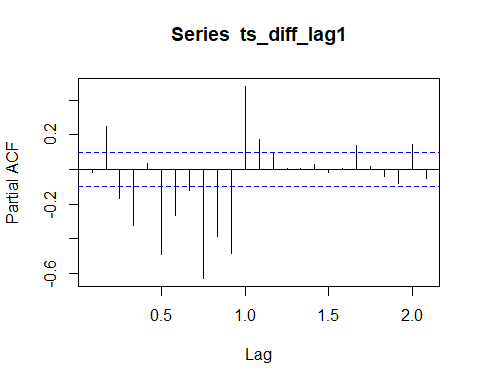
ts\_diff\_lag1 <- diff(ts, lag = 1)  
plot(ts\_diff\_lag1)



acf(ts\_diff\_lag1)



pacf(ts\_diff\_lag1)



Time series is not stationary after differencing with lag 1. There are obvious autocorrelations at lag 5, 10, 15 and 20. The pacf shows autocorrelations at lag 3, 5, 6, 8, 9, 10 and 11. It would be interesting to analyze the values at lag 5 and 10 in more detail. One should consider applying the log transformation as well.

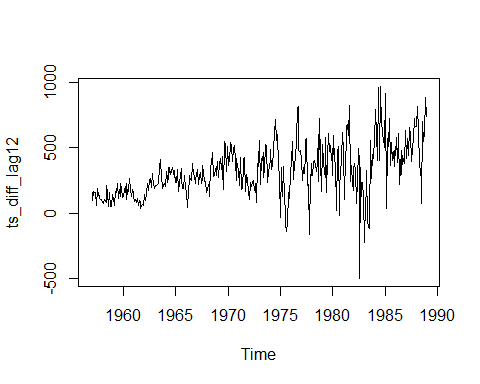
ts\_diff\_lag12 <- diff(ts, lag = 12)  
head(ts, 36)

## Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec  
## 1956 1254 1290 1379 1346 1535 1555 1655 1651 1500 1538 1486 1394  
## 1957 1409 1387 1543 1502 1693 1616 1841 1787 1631 1649 1586 1500  
## 1958 1497 1463 1648 1595 1777 1824 1994 1835 1787 1699 1633 1645

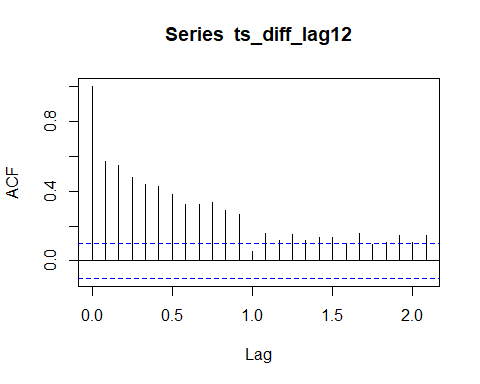
head(ts\_diff\_lag12, 24)

## Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec  
## 1957 155 97 164 156 158 61 186 136 131 111 100 106  
## 1958 88 76 105 93 84 208 153 48 156 50 47 145

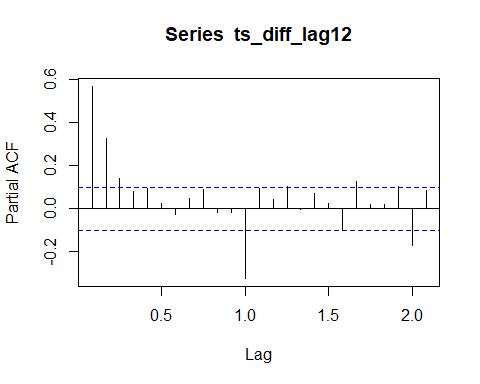
plot(ts\_diff\_lag12)



acf(ts\_diff\_lag12)



pacf(ts\_diff\_lag12)



Time series is not stationary after differencing with lag 12 as well. There are obvious autocorrelations at lag 1 to 9. The pacf shows autocorrelations at lag 1, 2, 10, 17 and 20. It would be interesting to analyze the values at lag 1 in more detail. One should consider applying the log transformation as well.

## Exercise 2.6

2.6 Adresse wird von Hr. Schütz nachgeliefert, PACF verwenden

# to do

# to do

## Exercise 2.7

Prozess / Modell hat die gleiche Formel wie 2.3

# to do

# to do

# to do