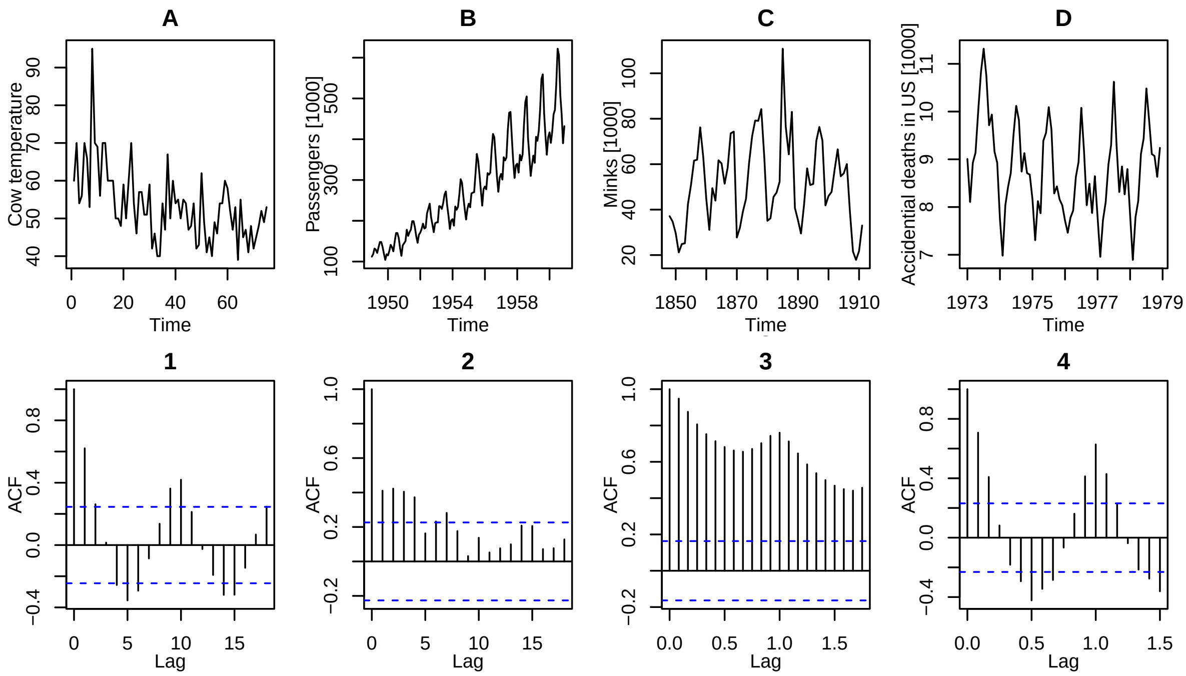
RTP Exercise Sheet

Series 3

Exercise 3.1

Below you find the plots and the correlograms of four datasets. The correlograms have been permutated. Please find for each data sets (A-D) the appropriate corellogram (1 - 4).



R-hints:

```
library(fma)
# Data set cow temperature
cowtemp
# Data set air passengers
AirPassengers
# Data set mink trappings
mink
# Data set accidential deaths in the US
usdeaths
```

Exercise 3.2

Let us now consider the electricity production of Australia in GWh in the period from January 1958 to December 1990. You may download the data from

`https://raw.githubusercontent.com/dallascard/Introductory_Time_Series_with_R_datasets/master/cbe.dat.`

The aim of this exercise is to compare the effect of different algorithms to decompose a time series representation in trend, seasonality and remainder by means of their (partial) autocorrelation function.

a) Start by considering the plot of the time series. Why is not meaningful to interpret the correlogram of this time series?

   Explain in a few sentences.

b) Decompose the time series into trend, seasonal component and remainder using the R function `decompose()`, which performs the decomposition with moving averages. Plot the remainder and its correlogram and interpret the plots in a few sentences.

   R-Hints:

```
# example for decompose function
decomp <- decompose(tselec, type = "multiplicative")

# example to calculate the plugin estimator of
# the autocorrelation function
acf(..., na.action = na.pass, plot = TRUE)
```

   The function employs a filter to estimate the trend; therefore, the first and the last few entries of the decomposition are not defined, i.e. the have the value `NA` in R. To prevent issues of R, the parameter `na.action = na.pass` (asking R to ignore `NA` entries) has to be employed.

c) Decompose the log-transformed time series using the R function `stl()`. Estimate the seasonal effect once by averaging over all years (parameter `s.window = "periodic"`) and once by choosing an appropriate smoothing window (parameter `s.window = ...`). Recall that the window length has to be odd. An appropriate smoothing window may be determined by the R-function `monthplot()`. For both estimation approaches (averaging and smoothing window), plot the remainder and its correlogram, and comment on the plots.

   R-hint:

```
elec.stl <- stl(log(tselec), s.window = ...)
```

d) Explain why you used the parameter `type = "multiplicative"` in Task b), and why you log-transformed the time series before performing an `stl()` decomposition in Task c).

e) As a last algorithm consider the differencing approach. Choose a lag of 1 and 12 (months) to eliminate a trend and periodic structures. Plot the resulting time series and autocorrelation function. Compare the results to the previous methods.

Exercise 3.3

(Optional:) In this exercise, we will calculate the lagged scatter plot and the plug-in estimator without employing the internal R function.

a) Write a function to calculate the lagged scatter plot estimator for the autocorrelation. For this, you may extend the code given in the lecture notes.

b) Develop a function to calculate the plug-in estimator for the autocorrelation.

c) Calculate the two estimates for the `beer` and the `chicken` dataset. The `beer` and the `chicken` dataset is contained in the "fma"package. In case it is not already loaded, one can load it with the command `library(fma)`.

Disclaimer: Parts of the exercises are adopted from 'Applied Time Series Analysis' course at ETHZ by Marcel Dettling.