Time Series (732A62) Lab1

Anubhav Dikshit(anudi287) and Maximilian Pfundstein(maxpf364)

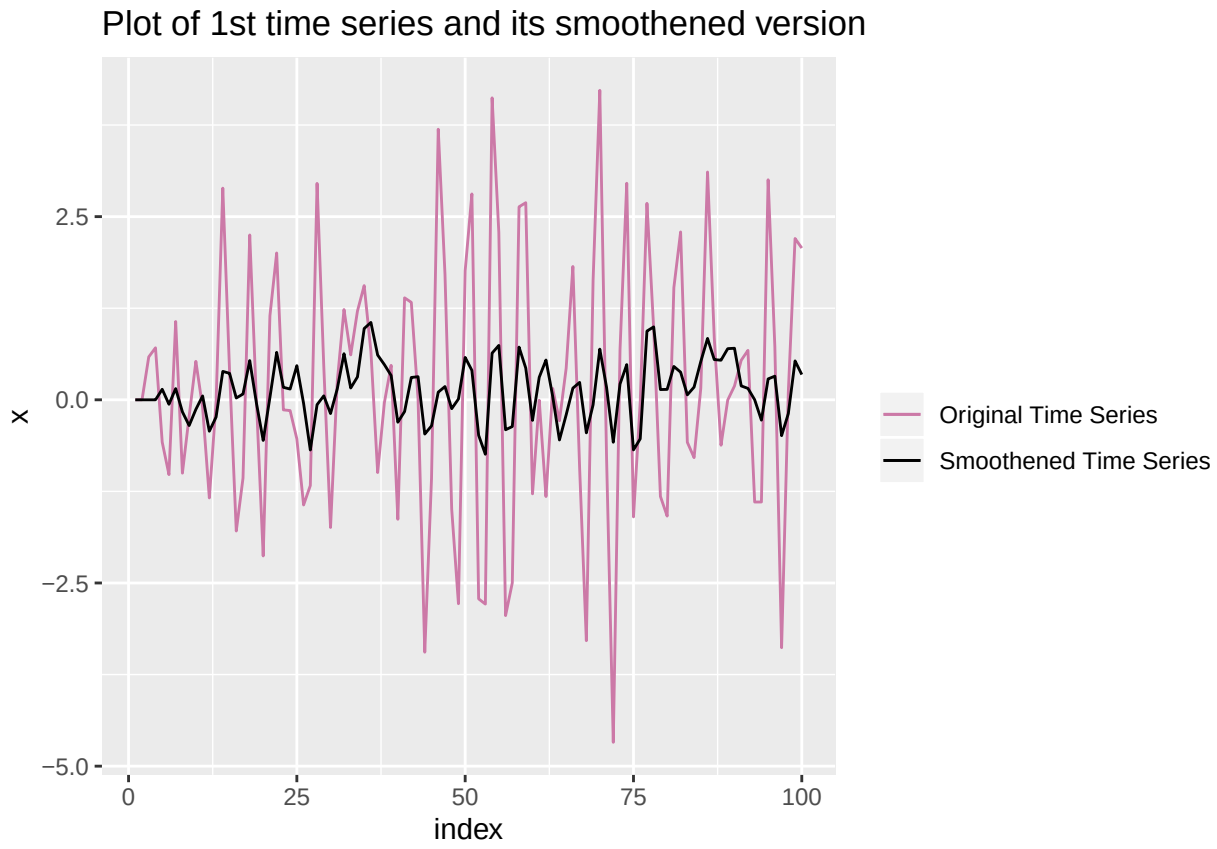
07 September, 2019

Contents

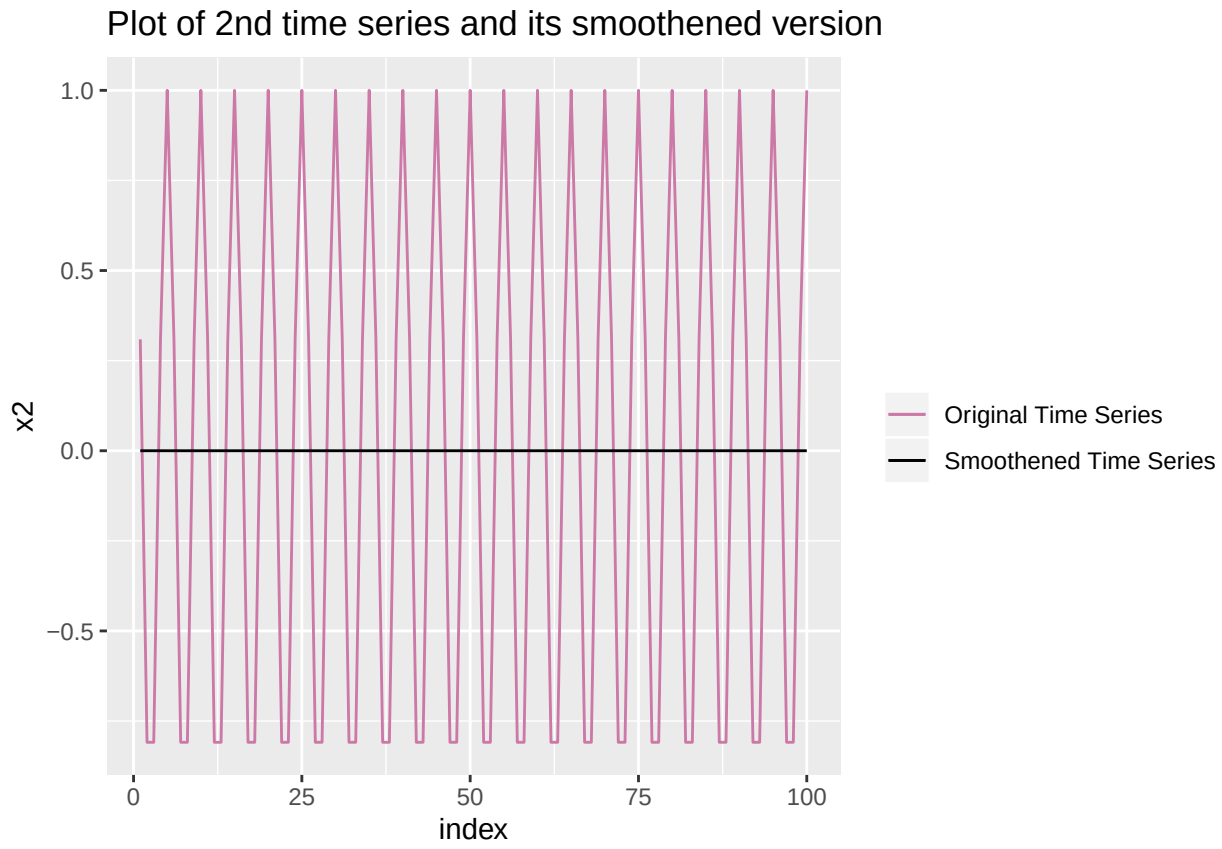
Assignment 1. Computations with simulated data

a) Generate two time series $x_t = -0.8x_{t-2} + w_t$, where $x_0 = x_1 = 0$ and $x_t = cos(\frac{2\pi t}{5})$ with 100 observations each. Apply a smoothing filter $v_t = 0.2(x_t + x_{t-1} + x_{t-2} + x_{t-3} + x_{t-4})$ to these two series and compare how the filter has affected them.

```
set.seed(12345)

n = 100
x <- vector(length = n)
x2 <- vector(length = n)

x[1] <- 0
x[2] <- 0

#first series generation
for(i in 3:n){
  x[i] <- -0.8 * x[i-2] + rnorm(1,0,1)
}

#second series generation
for(i in 1:n){
  x2[i] <- cos(0.4*pi*i)
}

# smoothing filter function
smoothing_filter <- function(x){
v <- vector(length = length(x))
for(i in 5:length(x)){
  v[i] = 0.2 * (x[i] + x[i-1] + x[i-2] + x[i-3] + x[i-4])
}
return(v)
}

#generate smoothed series
smooth_x <- smoothing_filter(x)
smooth_x2 <- smoothing_filter(x2)

#adding everything to a dataframe
df <- cbind(x,x2,smooth_x,smooth_x2) %>% as.data.frame() %>% mutate(index=1:100)

ggplot(df, aes(x=index)) +
  geom_line(aes(y=x, color="Original Time Series")) +
  geom_line(aes(y=smooth_x, color="Smoothened Time Series")) +
  ggtitle("Plot of 1st time series and its smoothened version") +
    scale_colour_manual("", breaks = c("Original Time Series", "Smoothened Time Series"),
                        values = c("#CC79A7", "#000000"))
```




```
ggplot(df, aes(x=index)) +
  geom_line(aes(y=x2, color="Original Time Series")) +
  geom_line(aes(y=smooth_x2, color="Smoothened Time Series")) +
  ggtitle("Plot of 2nd time series and its smoothened version") +
    scale_colour_manual("", breaks = c("Original Time Series", "Smoothened Time Series"),
                        values = c("#CC79A7", "#000000"))
```



b) Consider time series $x_t - 4x_{t-1} + 2x_{t-2} + x_{t-5} = w_t + 3w_{t-2} + w_{t-4} - 4w_{t-6}$. Write an appropriate R code to investigate whether this time series is casual and invertible.

Causality: ARMA(p,q) is causal iff roots $\phi(z') = 0$ are outside unit circle. eg: $x_t = 0.4x_{t-1} + 0.3x_{t-2} + w_t$, roots are -> $1 - 0.4B + 0.3B^2$

equation is: $\phi(Z) = 1 - 4B + 2B^2 + 0B^3 + 0B^4 + B^5$

```
z = c(1,-4,2,0,0,1)
polyroot(z)
```

```
## [1]  0.2936658+0.000000i -1.6793817+0.000000i  1.0000000-0.000000i
## [4]  0.1928579-1.410842i  0.1928579+1.410842i
```

```
any(Mod(polyroot(z))<1)
```

```
## [1] TRUE
```

Invertible: ARMA(p,q) is causal iff roots $\theta(z') = 0$ are outside unit circle.

equation is: $\theta(Z) = 1 + 3B^2 + B^4 - 4B^6$

```
z = c(1,3,0,1,0,-4)
polyroot(z)
```

```
## [1] -0.3266733+0.0000000i -0.9084665-0.0000000i  0.0870912-0.8867996i
## [4]  1.0609573+0.0000000i  0.0870912+0.8867996i
```

```r
any(Mod(polyroot(z))<1)
```

```
## [1] TRUE
```

Analysis: Baring one of the roots all are inside the unit circle. Thus the time series is not invertiable.

c) Use built-in R functions to simulate 100 observations from the process $x_t + \frac{3}{4}x_{t-1} = w_t - \frac{1}{9}w_{t-2}$, compute sample ACF and theoretical ACF, use seed 54321. Compare the ACF plots.

```r
set.seed(54321)

series <- arima.sim(n = 100, list(ar = c(-3/4), ma = c(0,-1/9)))
```

Appendix

```r
knitr::opts_chunk$set(echo = TRUE)
options(scipen=999)

library("ggplot2")
library("tidyverse")
library("gridExtra") # combine plots
library("knitr")


# The palette with black:
cbbPalette <- c("#000000", "#E69F00", "#56B4E9", "#009E73",
                "#F0E442", "#0072B2", "#D55E00", "#CC79A7")
set.seed(12345)
set.seed(12345)

n = 100
x <- vector(length = n)
x2 <- vector(length = n)

x[1] <- 0
x[2] <- 0

#first series generation
for(i in 3:n){
  x[i] <- -0.8 * x[i-2] + rnorm(1,0,1)
}

#second series generation
for(i in 1:n){
  x2[i] <- cos(0.4*pi*i)
}

# smoothing filter function
smoothing_filter <- function(x){
v <- vector(length = length(x))
```

```
for(i in 5:length(x)){
  v[i] = 0.2 * (x[i] + x[i-1] + x[i-2] + x[i-3] + x[i-4])
}
return(v)
}

#generate smoothed series
smooth_x <- smoothing_filter(x)
smooth_x2 <- smoothing_filter(x2)

#adding everything to a dataframe
df <- cbind(x,x2,smooth_x,smooth_x2) %>% as.data.frame() %>% mutate(index=1:100)

ggplot(df, aes(x=index)) +
  geom_line(aes(y=x, color="Original Time Series")) +
  geom_line(aes(y=smooth_x, color="Smoothened Time Series")) +
  ggtitle("Plot of 1st time series and its smoothened version") +
    scale_colour_manual("", breaks = c("Original Time Series", "Smoothened Time Series"),
                        values = c("#CC79A7", "#000000"))

ggplot(df, aes(x=index)) +
  geom_line(aes(y=x2, color="Original Time Series")) +
  geom_line(aes(y=smooth_x2, color="Smoothened Time Series")) +
  ggtitle("Plot of 2nd time series and its smoothened version") +
    scale_colour_manual("", breaks = c("Original Time Series", "Smoothened Time Series"),
                        values = c("#CC79A7", "#000000"))

z = c(1,-4,2,0,0,1)
polyroot(z)
any(Mod(polyroot(z))<1)
z = c(1,3,0,1,0,-4)
polyroot(z)
any(Mod(polyroot(z))<1)
set.seed(54321)

series <- arima.sim(n = 100, list(ar = c(-3/4), ma = c(0,-1/9)))
```