

# Lista 2 - estymacja i eliminacja trendu

Analiza szeregów czasowych

Karol Pustelnik 249828 i Jan Solarz

2021-03-02

## Spis treści

<b>1</b>	<b>Wstępne przekształcenie danych</b>	<b>2</b>
<b>2</b>	<b>Nieparametryczna estymacja trendu</b>	<b>4</b>
2.1	Różnicowanie . . . . .	4
2.2	Wyglądanie metodą ruchomej średniej . . . . .	6
2.3	Wyglądanie filtrem Spencera . . . . .	9
2.4	Wyglądanie wykładnicze . . . . .	10
2.5	Podsumowanie . . . . .	13
<b>3</b>	<b>Parametryczna estymacja trendu</b>	<b>13</b>
<b>4</b>	<b>Wstępna identyfikacja modelu</b>	<b>13</b>
<b>5</b>	<b>Bibliografia</b>	<b>13</b>

```

losuj.dekade <- function(album)
{
  set.seed(album)
  dekady <- seq(from=30, to=70, by=10)
  sample(dekady, 1)
}

```

*# Przykłady*

```
losuj.dekade(249828 + 243889)
```

```
## [1] 40
```

## 1 Wstępne przekształcenie danych

Na początku zastosujemy transformację Boxa z automatycznie dobranym parametrem `lambda`

```

library(xts)
library(forecast)

dane <- read.table("Dow_30-39.html", blank.lines.skip = T,
                  fill = T, sep="\t",
                  header=T, skip=2)
dane <- dane[-nrow(dane),] # usuwamy ostatni wiersz (znacznik </PRE>)
kurs <- dane$closing.values
daty <- as.POSIXlt(paste0("19",as.character(dane$Date)), format = "%Y%m%d")

# obiekt klasy 'xts' (dokładne daty notowań)
szereg.xts <- xts(kurs, order.by=daty)
szereg.ts <- ts(kurs)
autoplot(szereg.ts)

```



```
Box.test(szereg.ts) #test białosumowosci przed zastosowaniem różnicowania
```

```
##
```

```
## Box-Pierce test
```

```
##
```

```
## data: szereg.ts
```

```
## X-squared = 2977.5, df = 1, p-value < 2.2e-16
```

```
#i transformacji - p<0,05 zatem występuje statystycznie istotna korelacja
```

```
lambda<-BoxCox.lambda(szereg.ts) #wyznaczam najlepsza lambde do transformacji
```

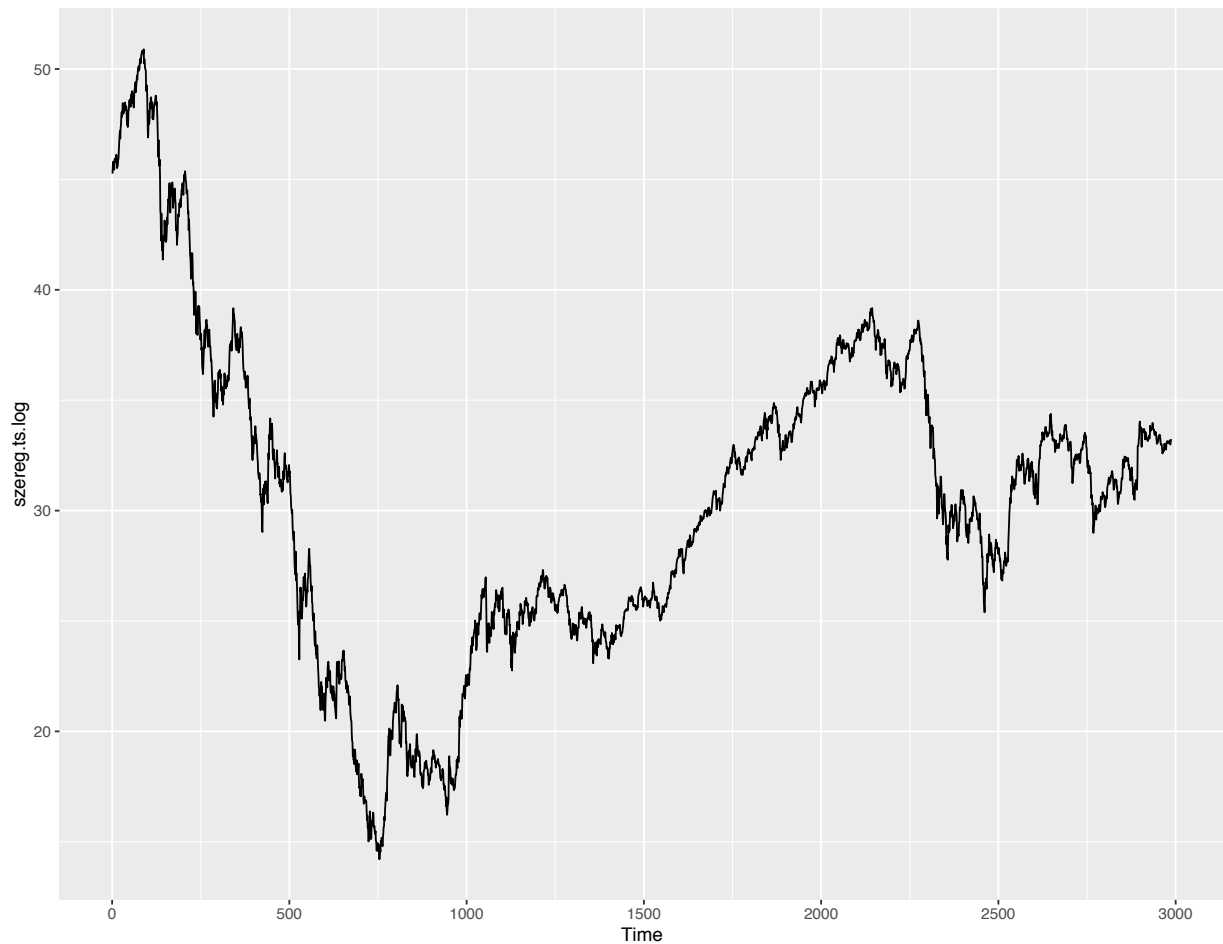
```
szereg.ts.log<-BoxCox(szereg.ts, lambda = lambda) # transformuje szereg dla  
#danej lambdy
```

## 2 Nieparametryczna estymacja trendu

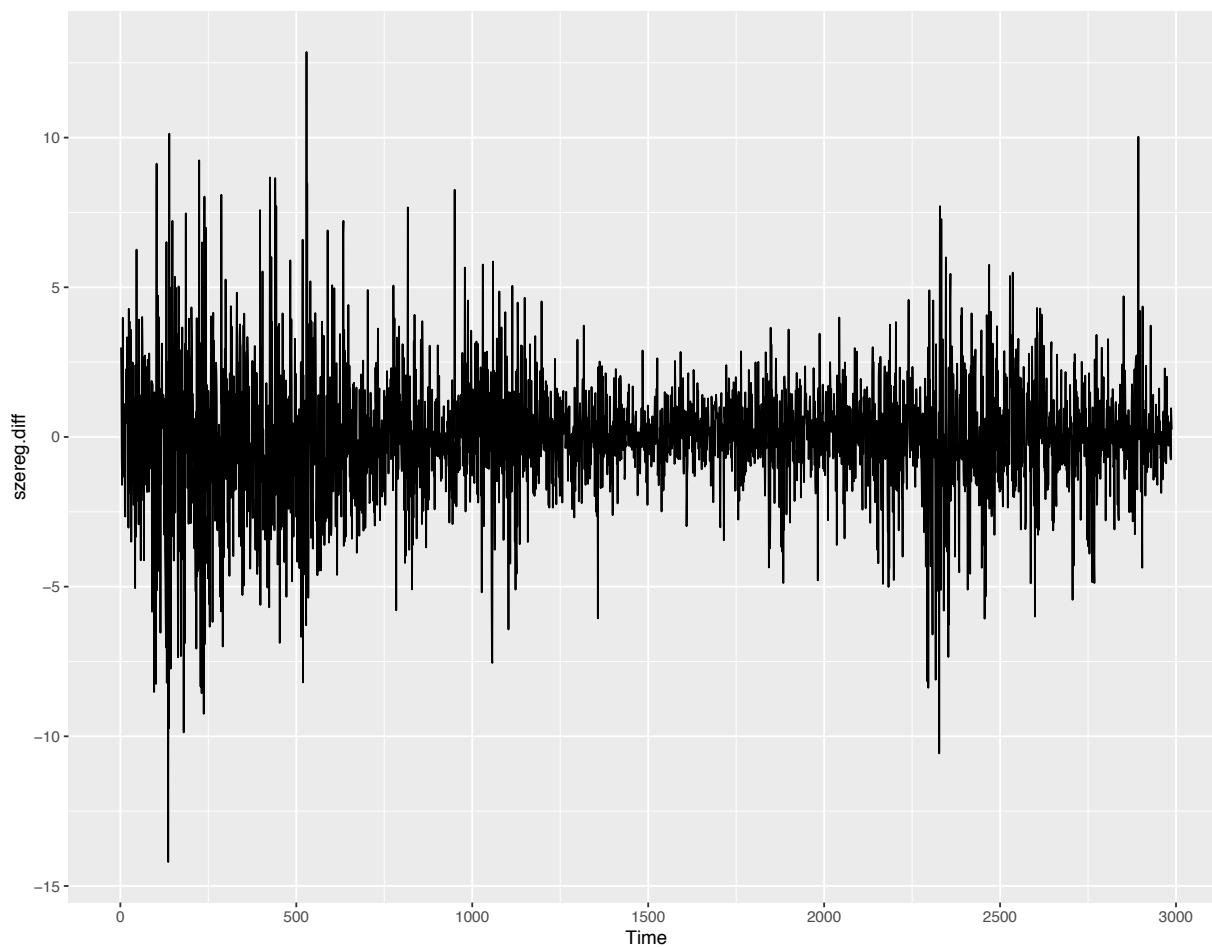
### 2.1 Różnicowanie

Dla naszych danych zastosujemy różnicowanie z opóźnieniem 1.

```
autoplot(szereg.ts.log)
```



```
szereg.diff<-diff(szereg.ts,differences =1)#różnicuje szereg z opóźnieniem 1  
#i różnicą 1 #testuję czy mamy biały szum  
autoplot(szereg.diff)
```



```
Box.test(szereg.diff)#testuję czy mamy biały szum -  $p > 0,05$  -
```

```
##
## Box-Pierce test
##
## data:  szereg.diff
## X-squared = 0.91411, df = 1, p-value = 0.339
#nie ma podstaw do odrzucenia hipotezy zerowej -
#raczej mamy do czynienia z białym szumem.
# Ztem dla różnicowania
```

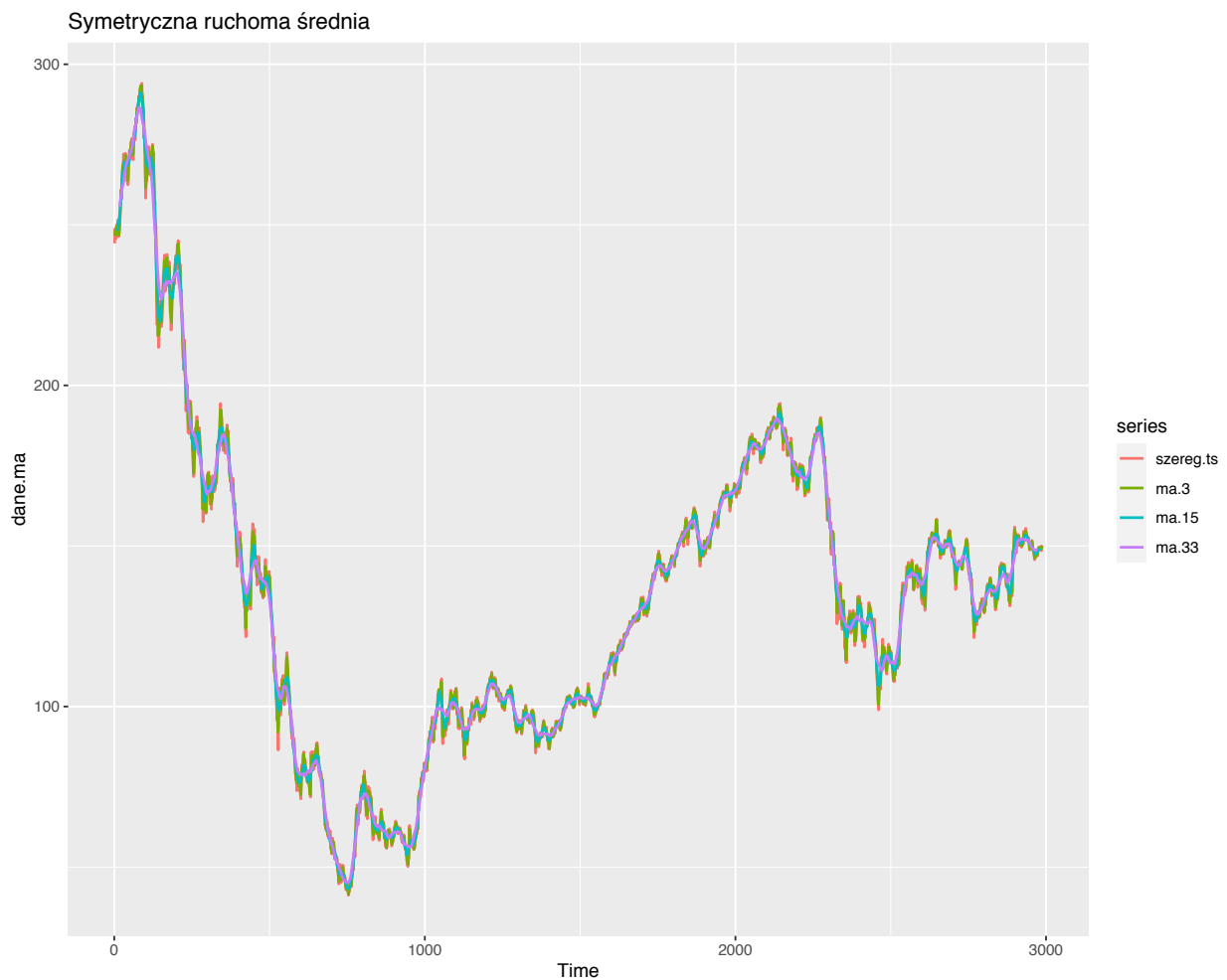
### 2.1.1 Wnioski

- Po zróżnicowaniu z opóźnieniem 1 dostaliśmy szereg stacjonarny - test białoszumowości nie nakazuje odrzucenia hipotezy zerowej. Z wykresu widać, że otrzymaliśmy porządkany efekt. Zatem dalej nie kontynuujemy różnicowania

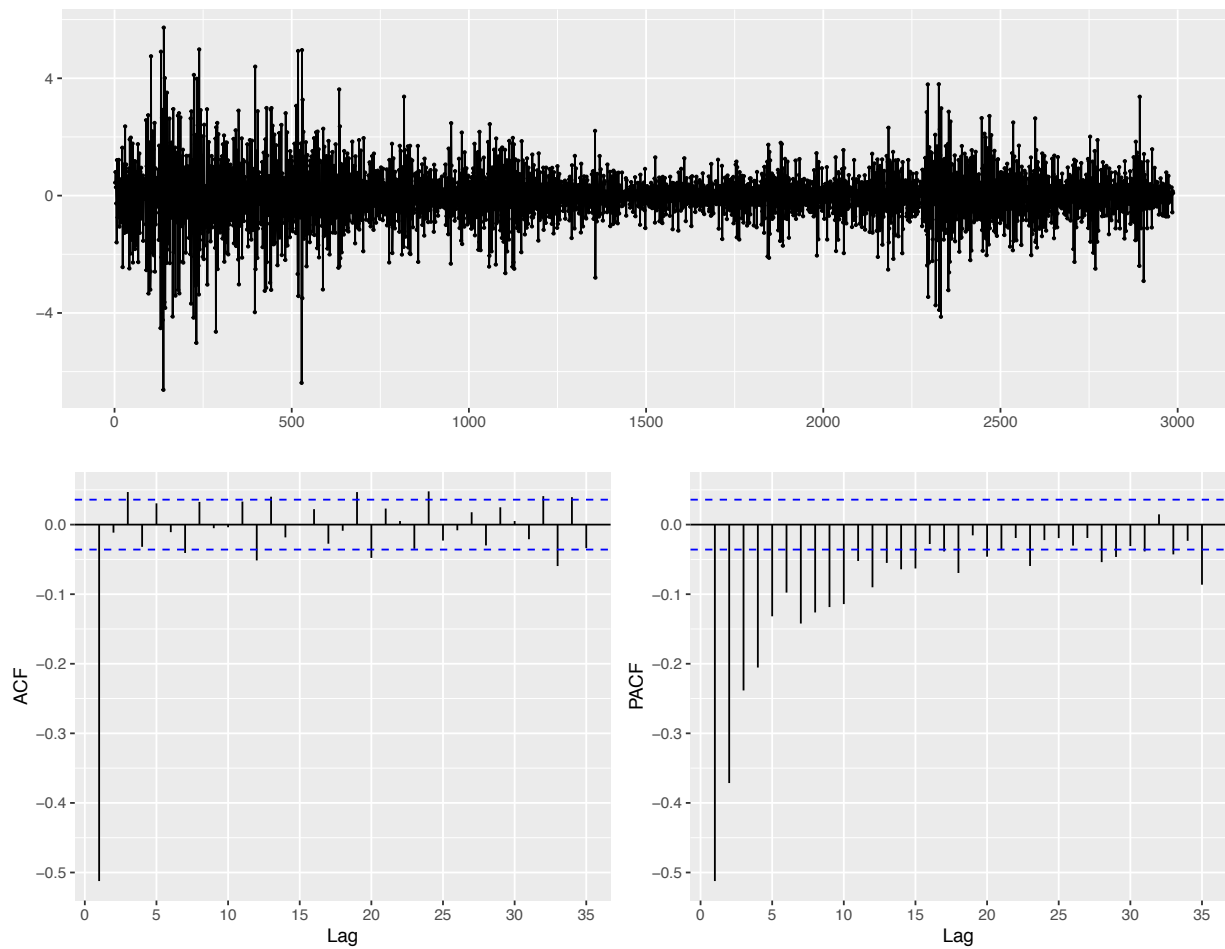
## 2.2 Wygładzanie metodą ruchomej średniej

Zastosujemy metodę wygładzania ruchomą średnią dla różnych parametrów  $q$  i porównamy otrzymane wyniki.

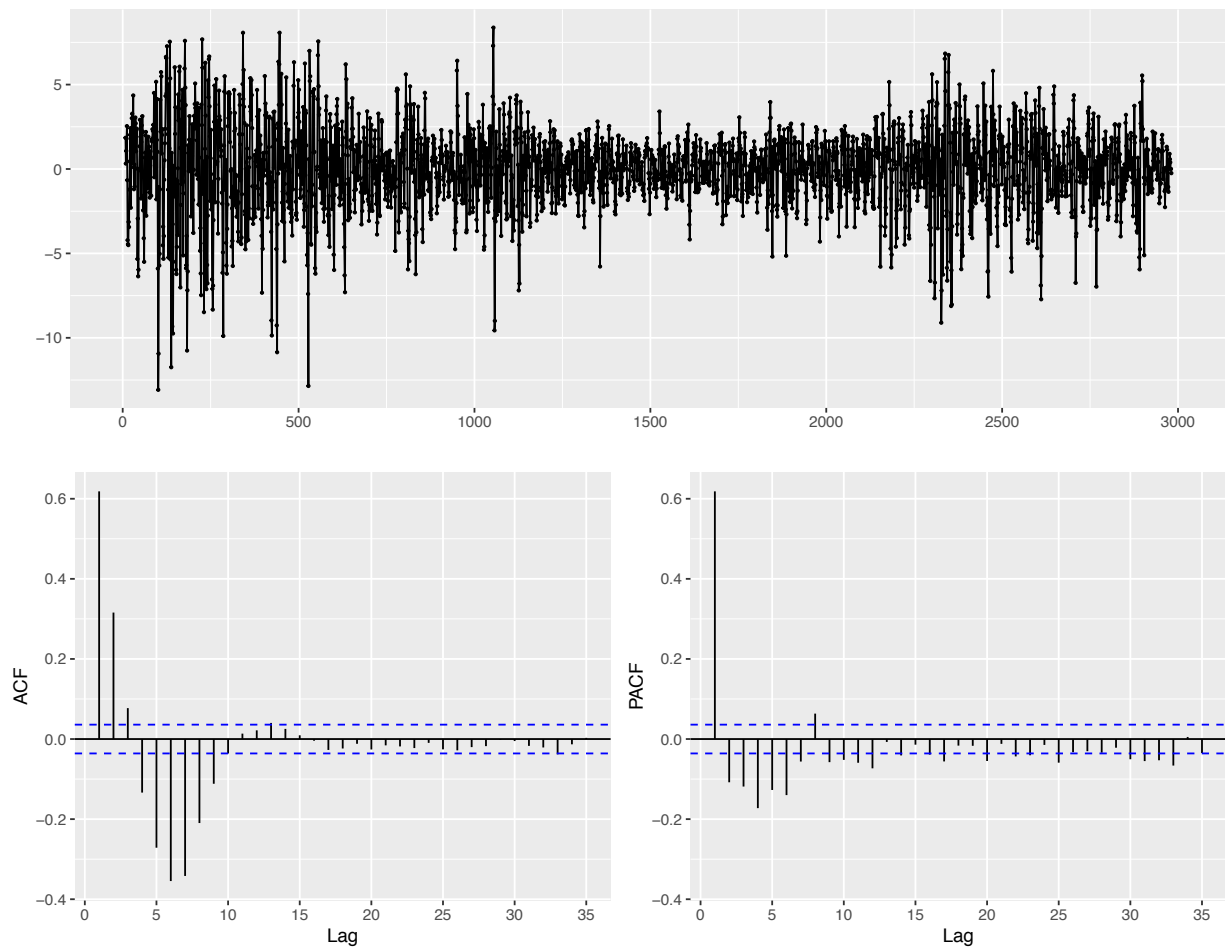
```
ma.3 <- ma(szereg.ts, order = 3, centre = TRUE)
ma.15 <- ma(szereg.ts, order = 15)
ma.33 <- ma(szereg.ts, order = 33)
dane.ma <- ts.union(szereg.ts, ma.3, ma.15, ma.33)
autoplot(dane.ma, main="Symetryczna ruchoma średnia", lwd=.75)
```



```
resid.ma.3 <- szereg.ts-ma.3 #Uwaga na NA!
ggtsdisplay(resid.ma.3)
```

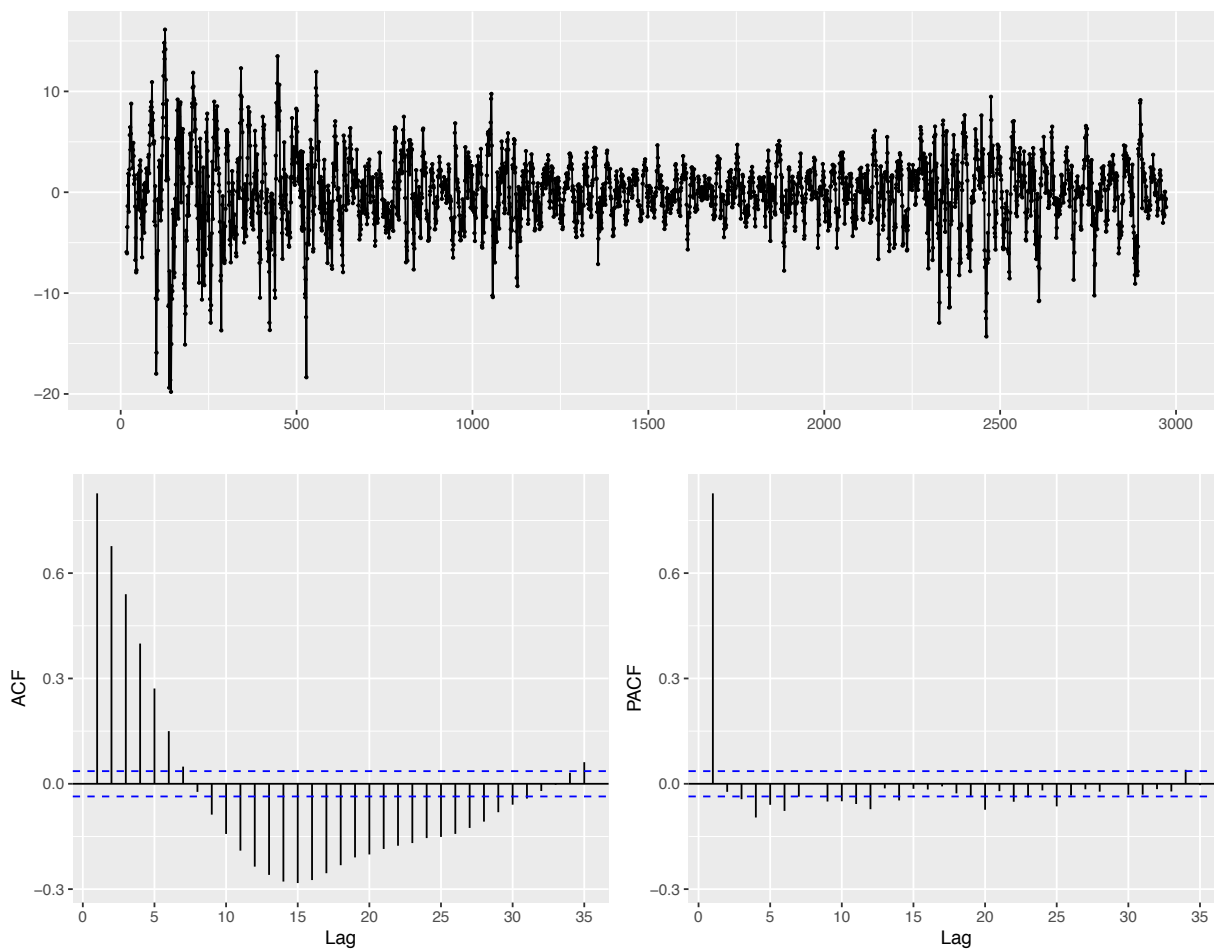


```
resid.ma.15 <- szereg.ts-ma.15 #Uwaga na NA!  
ggtsdisplay(resid.ma.15)
```



```
resid.ma.33 <- szereg.ts-ma.33 #Uwaga na NA!  
ggtstdisplay(resid.ma.33)
```





### 2.2.1 Wnioski

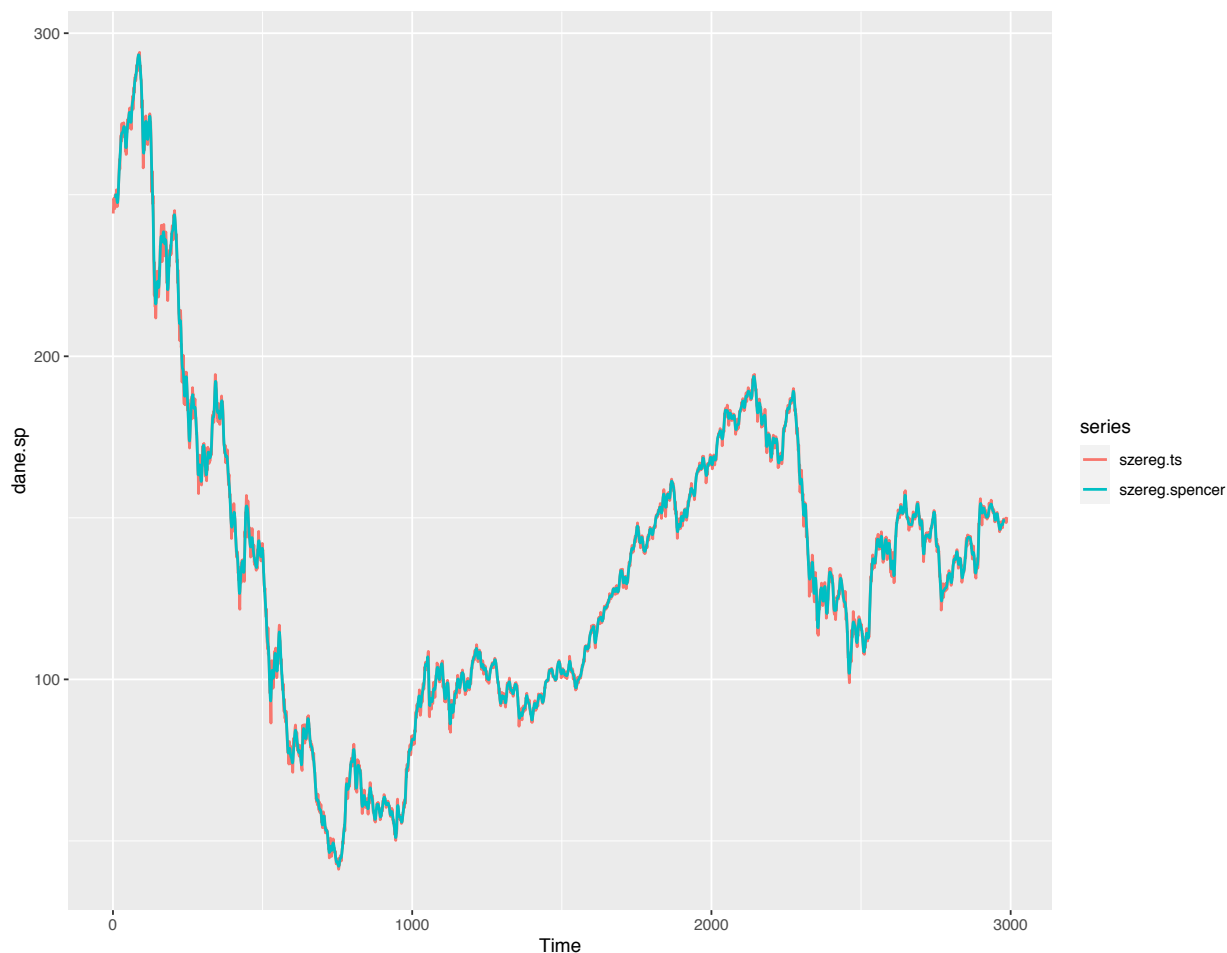
- Wraz ze wzrostem parametru  $q$  ( $\text{order} = 2q+1$ ) rośnie poziom wygładzenia
- Samo wygładzenie nie jest satysfukcjonujące - praktycznie nie zauważalne.

## 2.3 Wygładzanie filtrem Spencera

Zastosujemy teraz alternatywną metodę wygładzania ruchomą średnią - uwzględniając wagi dla obserwacji. Tutaj parametr  $q$  jest ustawiony na 7 ( $\text{order} = 15$ ).

```
library(signal)
szereg.spencer<-spencer(szereg.ts)

dane.sp <- ts.union(szereg.ts, szereg.spencer)
autoplot(dane.sp, lwd=0.75)
```



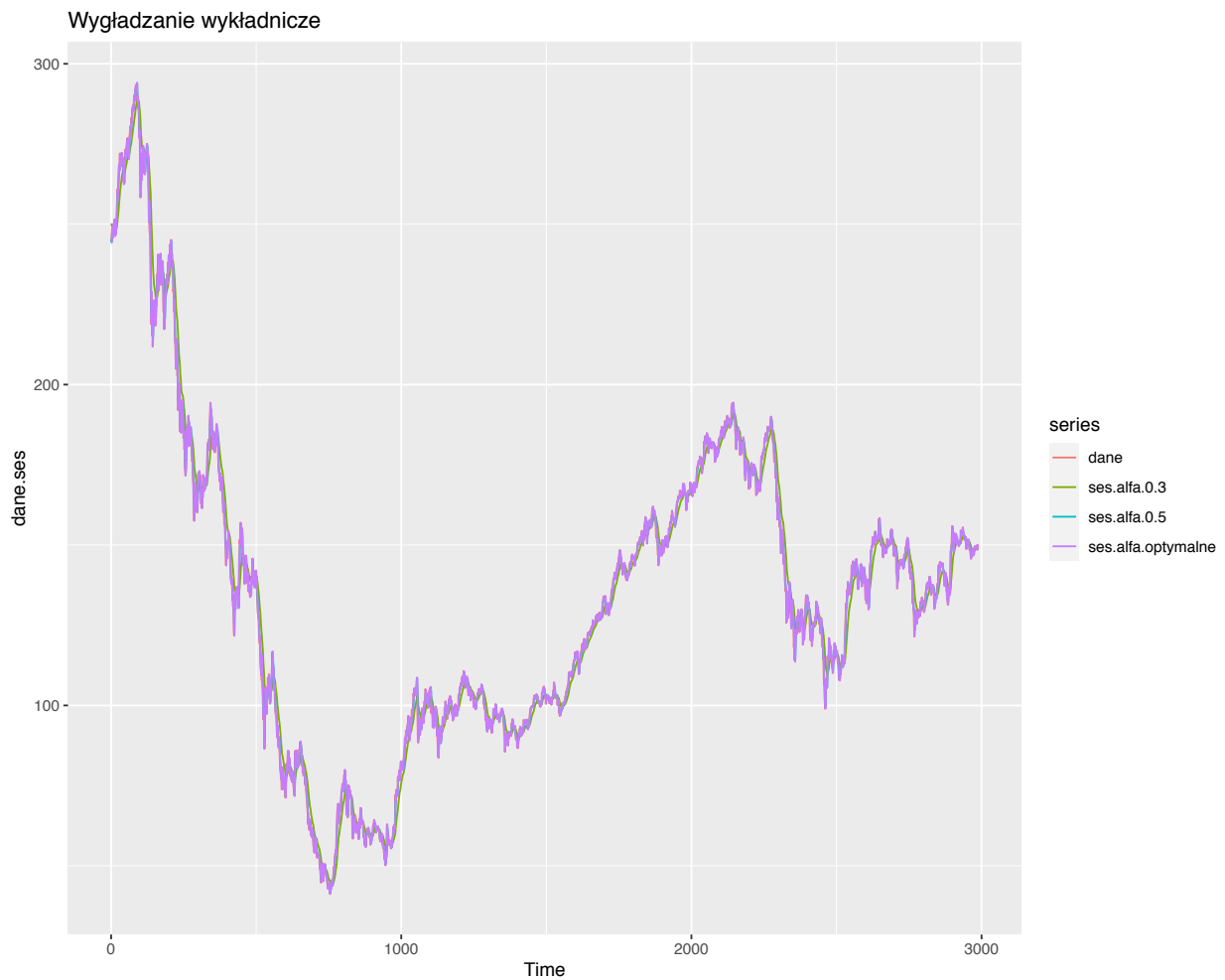
### 2.3.1 Wnioski

- Poziom wygładzenia jest mniejszy od najlepszego wygładzenia dla poprzedniej metody.

## 2.4 Wygładzanie wykładnicze

Kolejną metodą estymacji trendu jest wygładzanie wykładnicze. Zaprezentujemy wyniki dla różnych parametrów alfa.

```
library(expsmooth)
dane<-szereg.ts
ses.alfa03 <- ses(dane, alpha=0.1)$fitted
ses.alfa05 <- ses(dane, alpha=0.7)$fitted
ses.alfa.optim <- ses(dane)$fitted
dane.ses <- ts.union(dane, ses.alfa.0.3=sese.alfa03,
                    ses.alfa.0.5=sese.alfa05, ses.alfa.optymalne=sese.alfa.optim)
autoplot(dane.ses, main="Wygładzanie wykładnicze", lwd=.5)
```



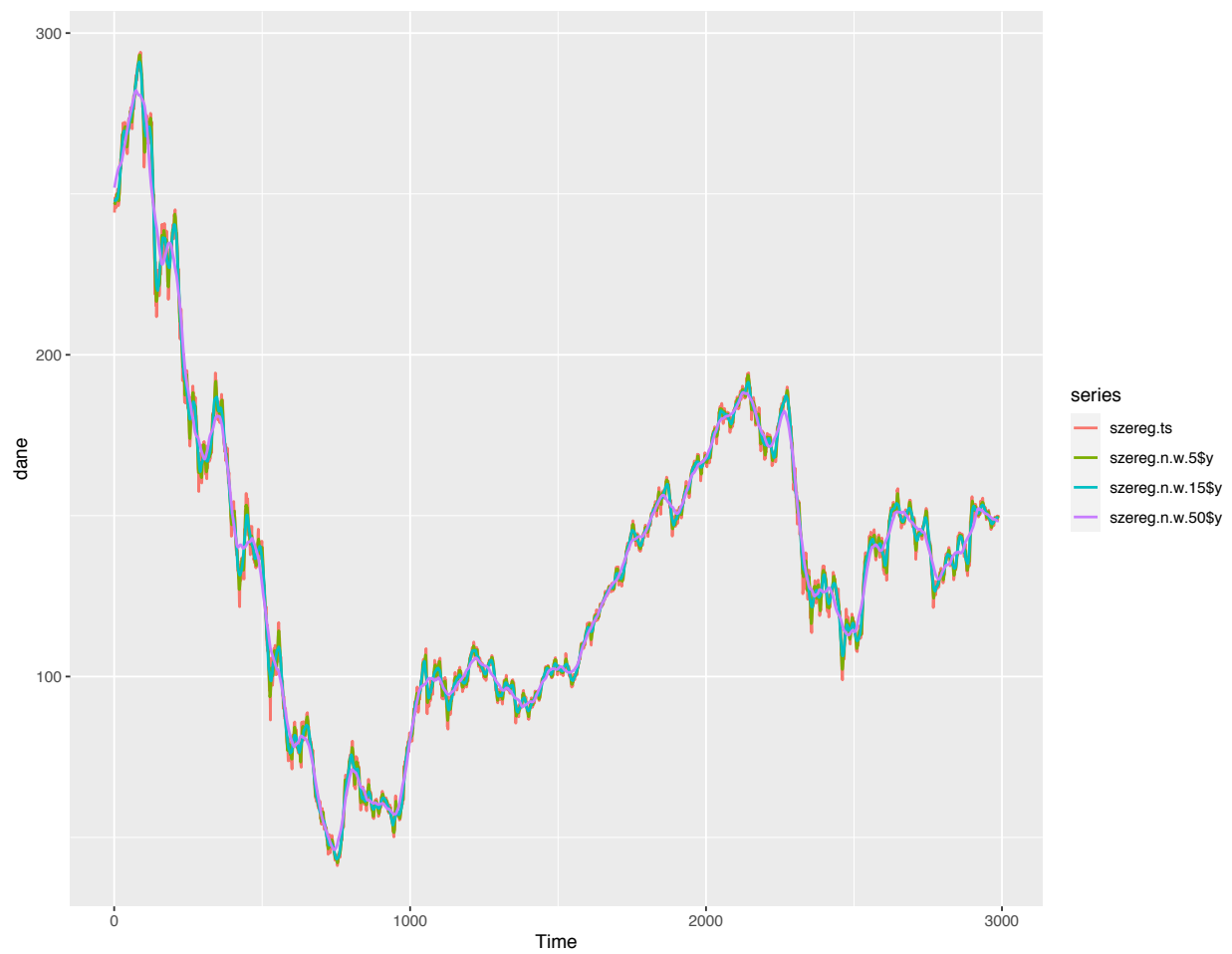
### 2.4.1 Wnioski

- Wyniki dla różnych alf się praktycznie nie różnią.
- Nie dostaliśmy satysfkującego wygładzenia.

```
length(szereg.ts)
```

```
## [1] 2988
```

```
x<-seq(1,2988,1)
szereg.n.w.5<-ksmooth(x, szereg.ts, bandwidth = 5)
szereg.n.w.15<-ksmooth(x, szereg.ts, bandwidth = 15)
szereg.n.w.50<-ksmooth(x, szereg.ts, bandwidth = 50)
dane <- ts.union(szereg.ts, szereg.n.w.5$y,szereg.n.w.15$y,szereg.n.w.50$y)
autoplot(dane,lwd=0.75)
```



### 2.4.2 Wnioski

- Dla szerszego okna dostaliśmy bardziej wygładzony wykres. Nadal nie jest on satysfukcjonujący - ciężko odczytać trend
- Wyniki są podobne jak dla metody ruchomej średniej

## 2.5 Podsumowanie

Nasze dane nie nadają się najlepiej do estymacji trendu, ponieważ jest on praktycznie nie zauważalny. Pomimo tego, najlepiej poradziły sobie metody:

- ruchomej średniej,
- jądrowa estymacja Nadaraya-Watsona.

Jeśli chodzi o eliminację “trendu” z naszych danych, to różnicowanie z opóźnieniem 1 dało nam porządkany biały szum.

## 3 Parametryczna estymacja trendu

## 4 Wstępna identyfikacja modelu

## 5 Bibliografia

### Literatura

- [1] Dr. Adam Zagdański, *Analiza szeregów czasowych, semestr letni 2021*
- [2] Robert H. Shumway, David S. Stoffer, *Time Series Analysis and Its Applications*, Springer 3rd Edition.