

PSTAT174-Project

Jon Carlos (3872520)

2023-03-13

```
library(tidyverse)
library(ggplot2)
library(ggfortify)
library(ggthemes)
library(zoo)
library(forecast)
library(Mcomp)
library(MASS)
library(fabletools)
library(reshape2)
```

```
EU_Max_Temperatures <- read.csv("Temperatures_In_Europe.csv")
names(EU_Max_Temperatures) <- c("date", "Berlin", "Paris", "Copenhagen", "Stockholm", "Rome")

head(EU_Max_Temperatures, 5)
```

```
##      date Berlin Paris Copenhagen Stockholm Rome
## 1 1900-01    8.4 12.7      6.2      5.0    NA
## 2 1900-02   17.4 16.9      6.2      4.5    NA
## 3 1900-03   15.0 16.9      7.6      5.5    NA
## 4 1900-04   24.1 25.8     19.2     14.0    NA
## 5 1900-05   29.4 28.9     23.4     21.0    NA
```

```
summary(EU_Max_Temperatures)
```

```
##      date      Berlin      Paris      Copenhagen
## Length:1473    Min.   : 1.50    Min.   : 6.10    Min.   : 1.50
## Class :character 1st Qu.:13.90    1st Qu.:16.00    1st Qu.:10.80
## Mode  :character Median :22.70    Median :23.72    Median :18.50
##              Mean   :21.92    Mean   :23.29    Mean   :18.35
##              3rd Qu.:29.90    3rd Qu.:29.80    3rd Qu.:25.90
```

```
##           Max.      :38.90   Max.      :42.61   Max.      :39.10
##
##   Stockholm           Rome
##   Min.      : 0.00   Min.      :14.60
##   1st Qu.: 9.00   1st Qu.:21.00
##   Median :16.75   Median :27.00
##   Mean    :16.95   Mean     :27.23
##   3rd Qu.:25.00   3rd Qu.:33.50
##   Max.     :41.00   Max.      :42.00
##   NA's     :3      NA's      :576
```

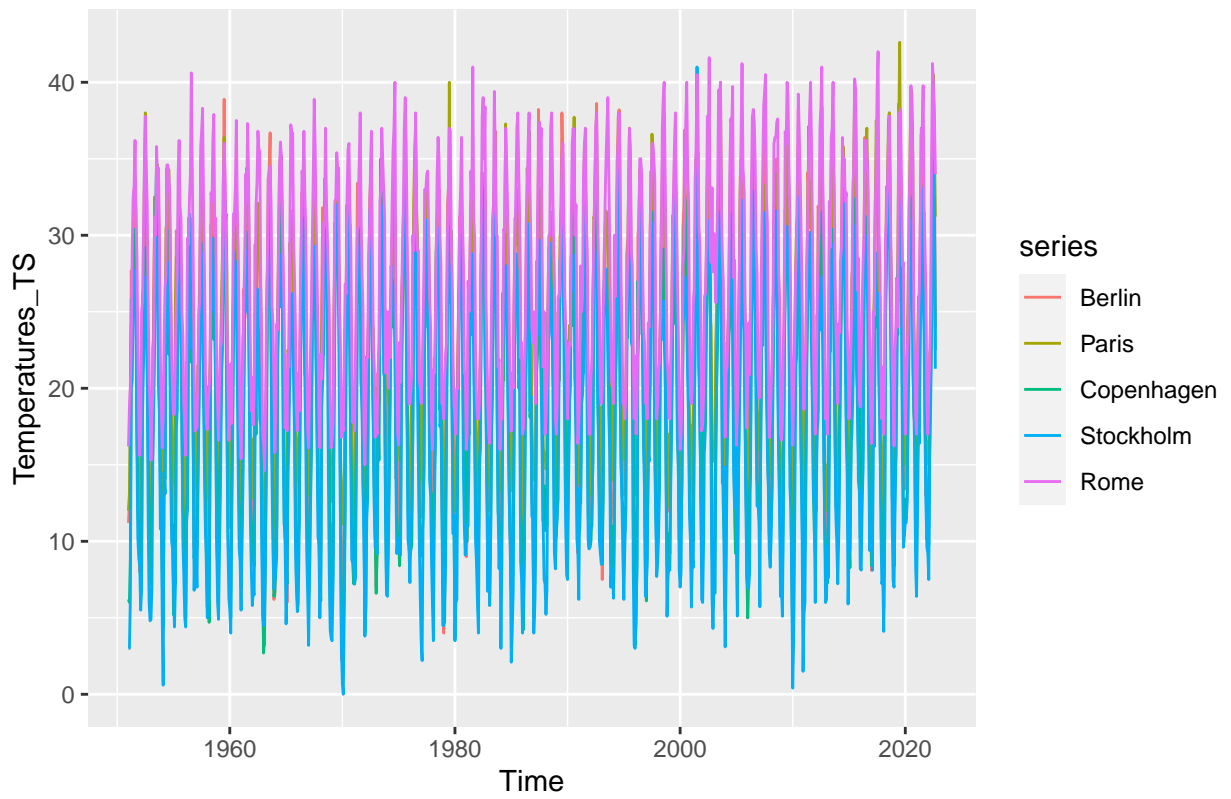
However, there is some missing data within the column for Rome as seen above so we must remove them. Because the data set is rather large to begin with starting all the way from the year 1900, we can get rid of data from each other column until Rome finally has some. Looking at the data set, Rome begins to have some data when `date = 1944-06`, but is still patchy with this information until around 1951-01. Therefore, all data from before this will be removed and only from 1951 and onward will be analyzed.

```
EU_Max_Temperatures <- EU_Max_Temperatures[EU_Max_Temperatures$date >= '1951-01',]
head(EU_Max_Temperatures, 5)
```

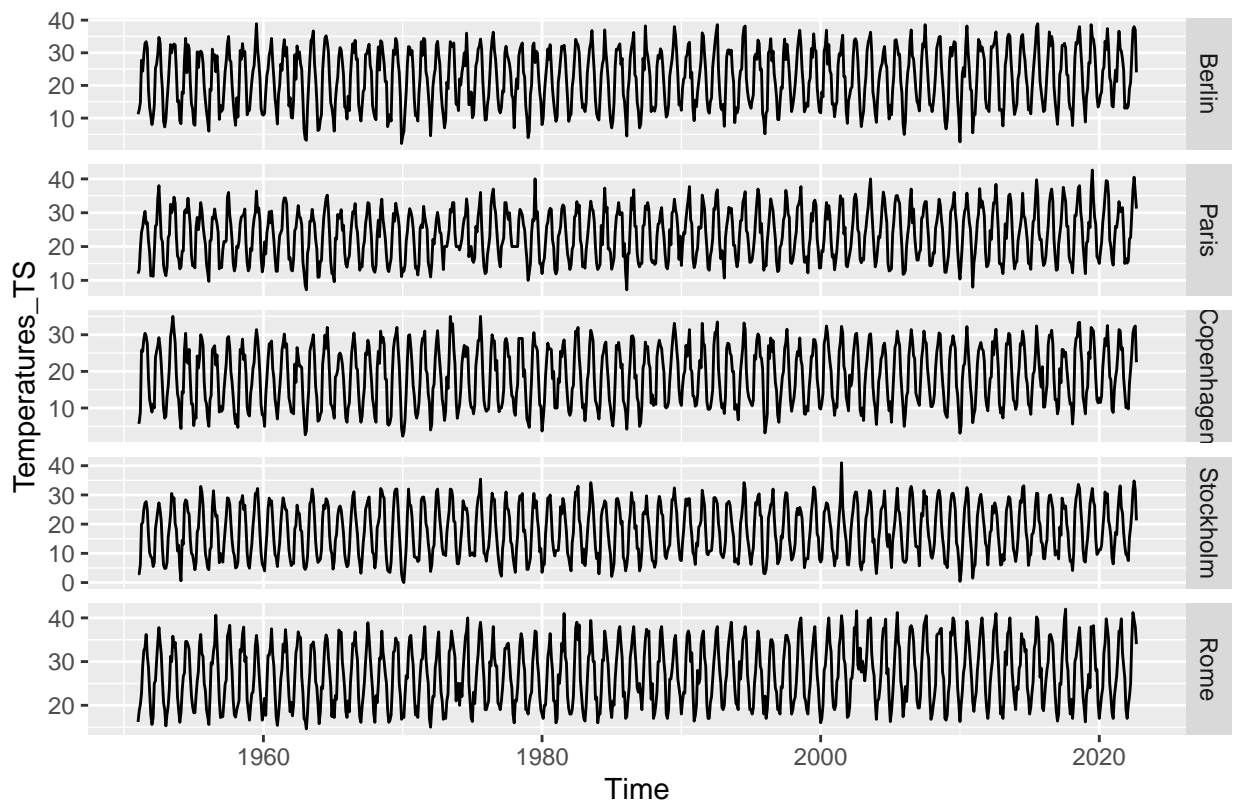
```
##      date Berlin Paris Copenhagen Stockholm Rome
## 613 1951-01   11.2  12.0      6.200      3.0 16.2
## 614 1951-02   12.5  13.1      6.000      3.0 18.7
## 615 1951-03   14.8  19.6      8.889      6.2 20.2
## 616 1951-04   27.7  24.2     25.800     20.0 22.8
## 617 1951-05   24.4  25.8     25.400     20.5 29.0
```

```
Temperatures_TS <- ts(EU_Max_Temperatures[,c(2:6)], start = c(1951), frequency = 12)

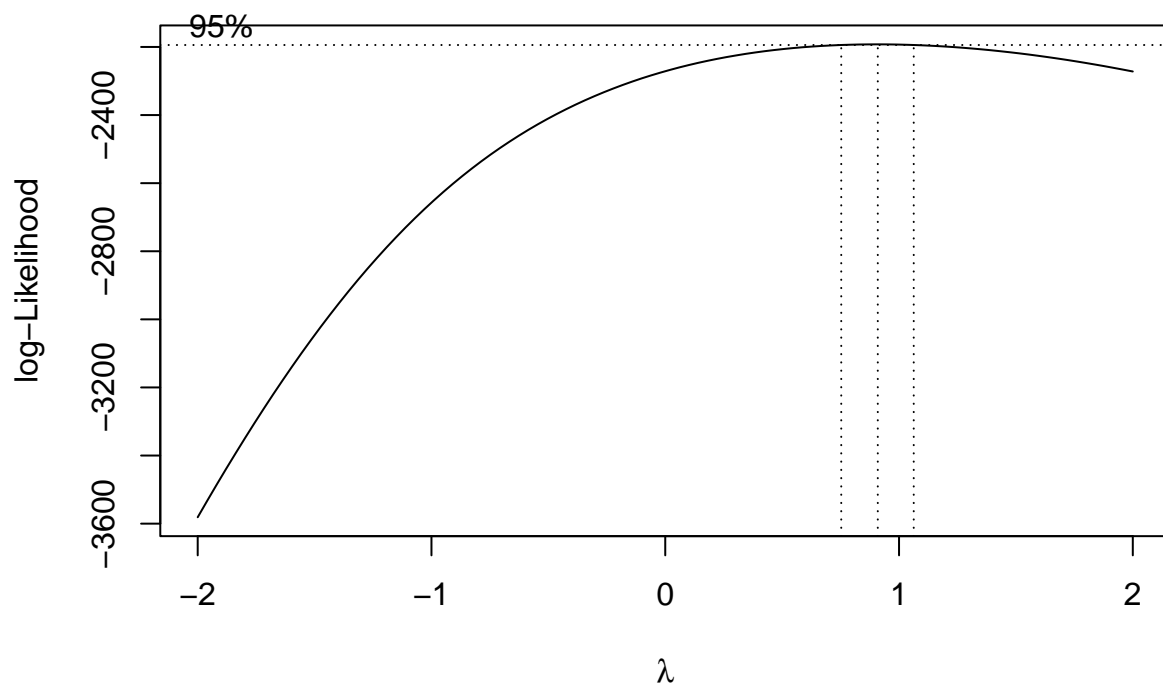
options(repr.plot.width =2, repr.plot.height =3)
autoplot(Temperatures_TS)
```



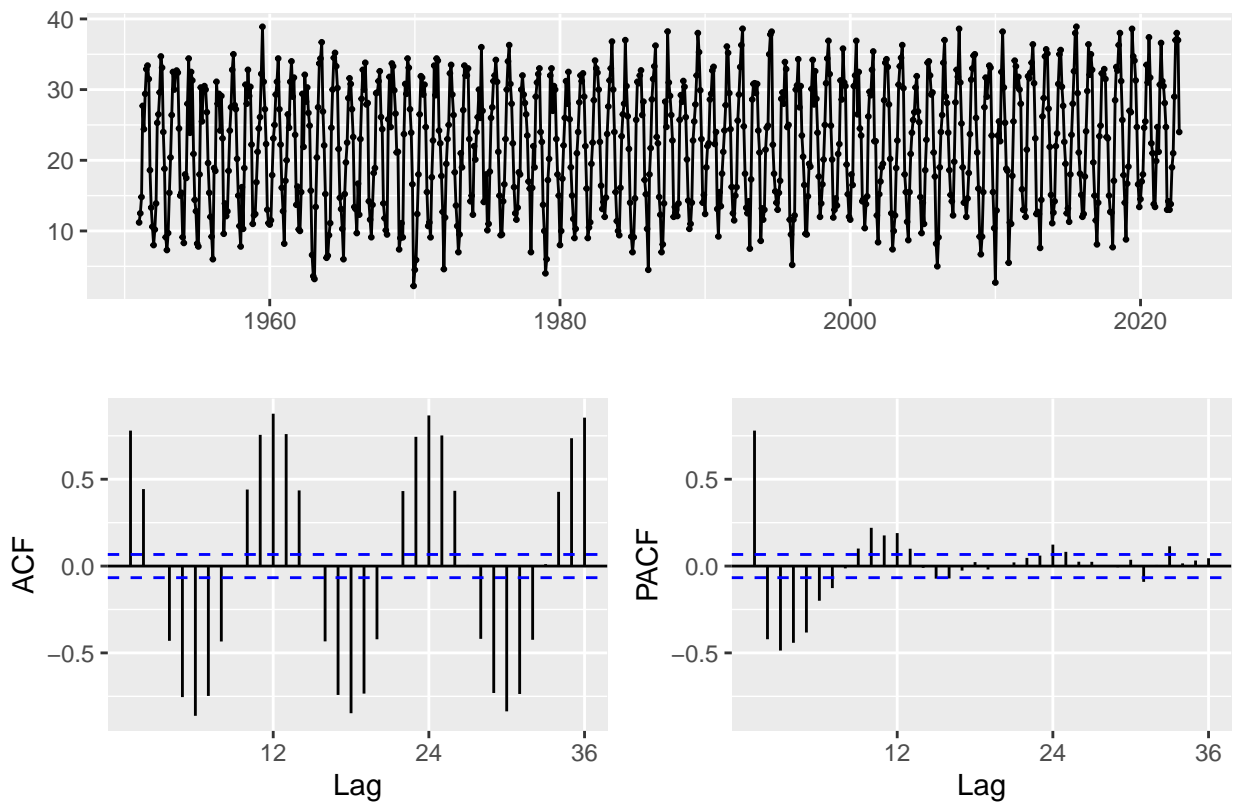
```
autoplot(Temperatures_TS, facets = TRUE)
```



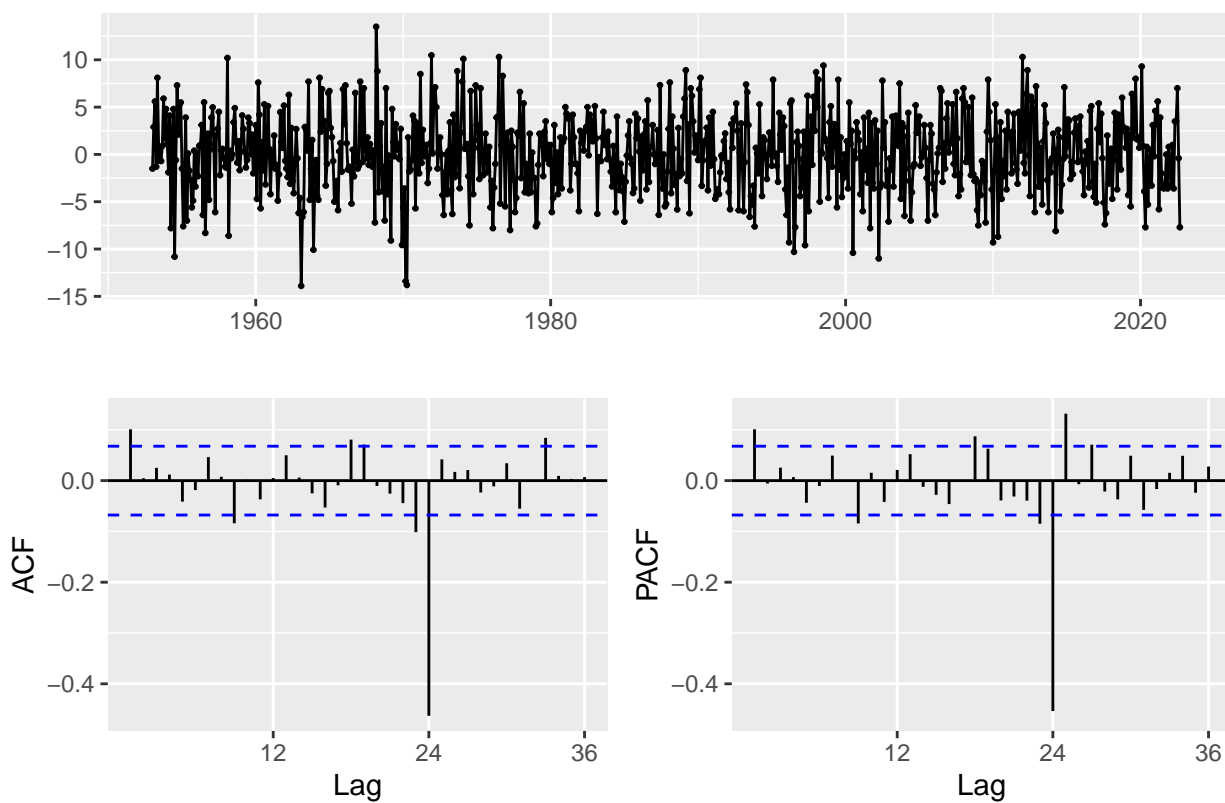
```
boxcox(lm(Temperatures_TS[,1] ~ 1))
```



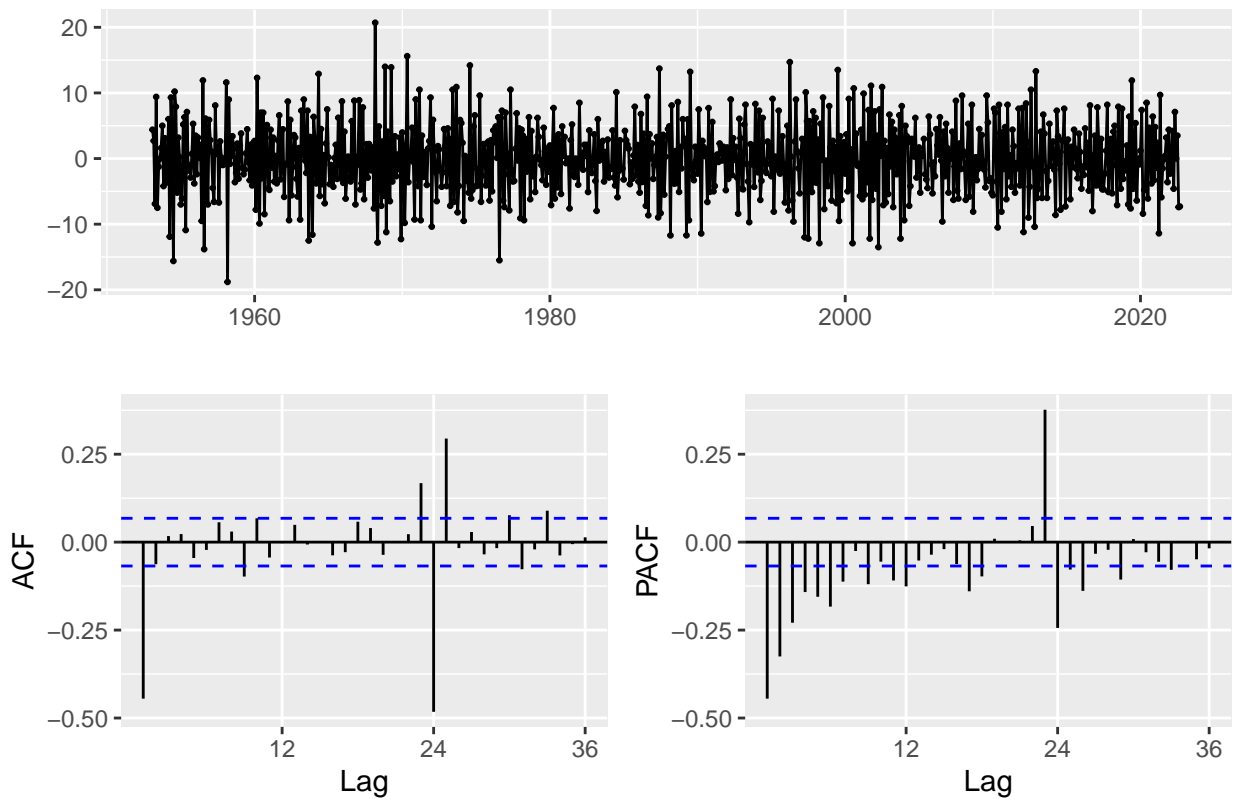
```
Temperatures_TS[,1] %>%  
  ggtsdisplay()
```



```
Temperatures_TS[,1] %>%
  diff(lag = 24) %>%
  ggtsdisplay()
```



```
Temperatures_TS[,1] %>%
  diff(lag = 24) %>% diff() %>%
  ggtsdisplay()
```

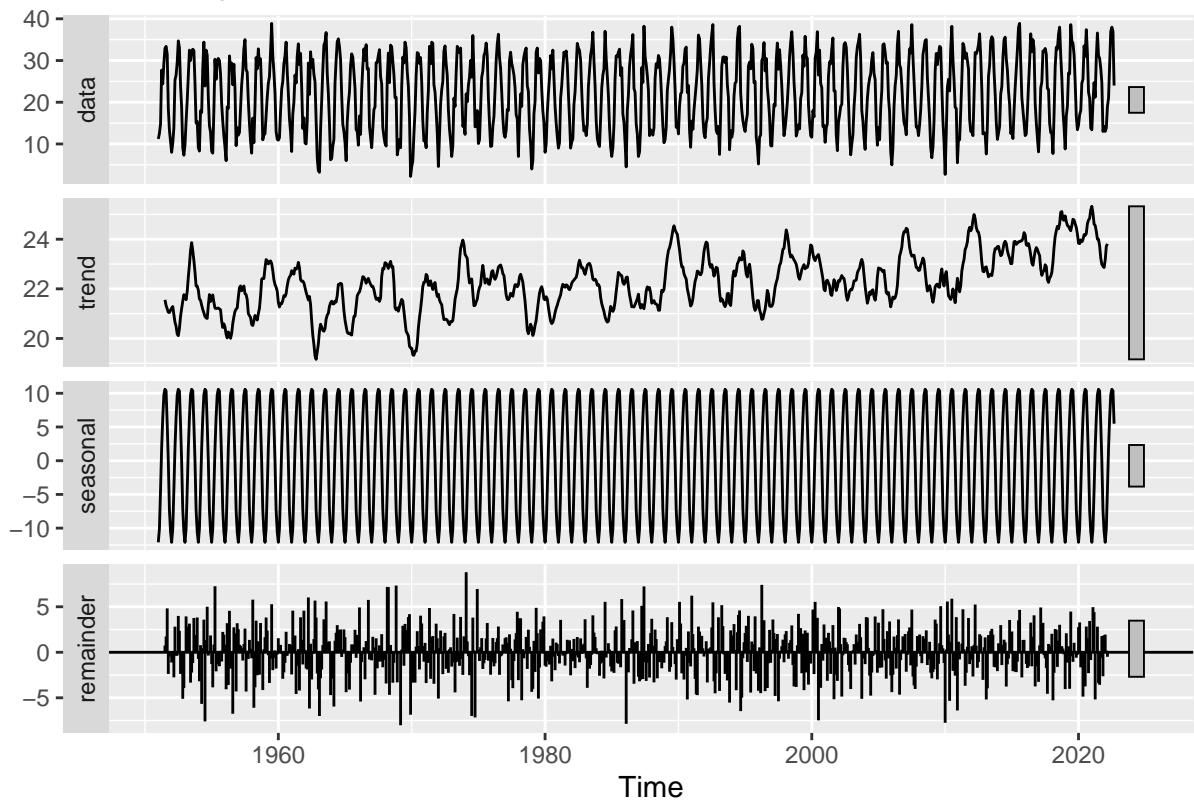


When rounding to the closest λ value, $\lambda = 1$. Therefore, we do not need to make any BoxCox transformations to the data.

```
Decomp_Temperatures_TS_Berlin <- decompose(Temperatures_TS[,1])
Decomp_Temperatures_TS_Paris <- decompose(Temperatures_TS[,2])
Decomp_Temperatures_TS_Copenhagen <- decompose(Temperatures_TS[,3])
Decomp_Temperatures_TS_Stockholm <- decompose(Temperatures_TS[,4])
Decomp_Temperatures_TS_Rome <- decompose(Temperatures_TS[,5])

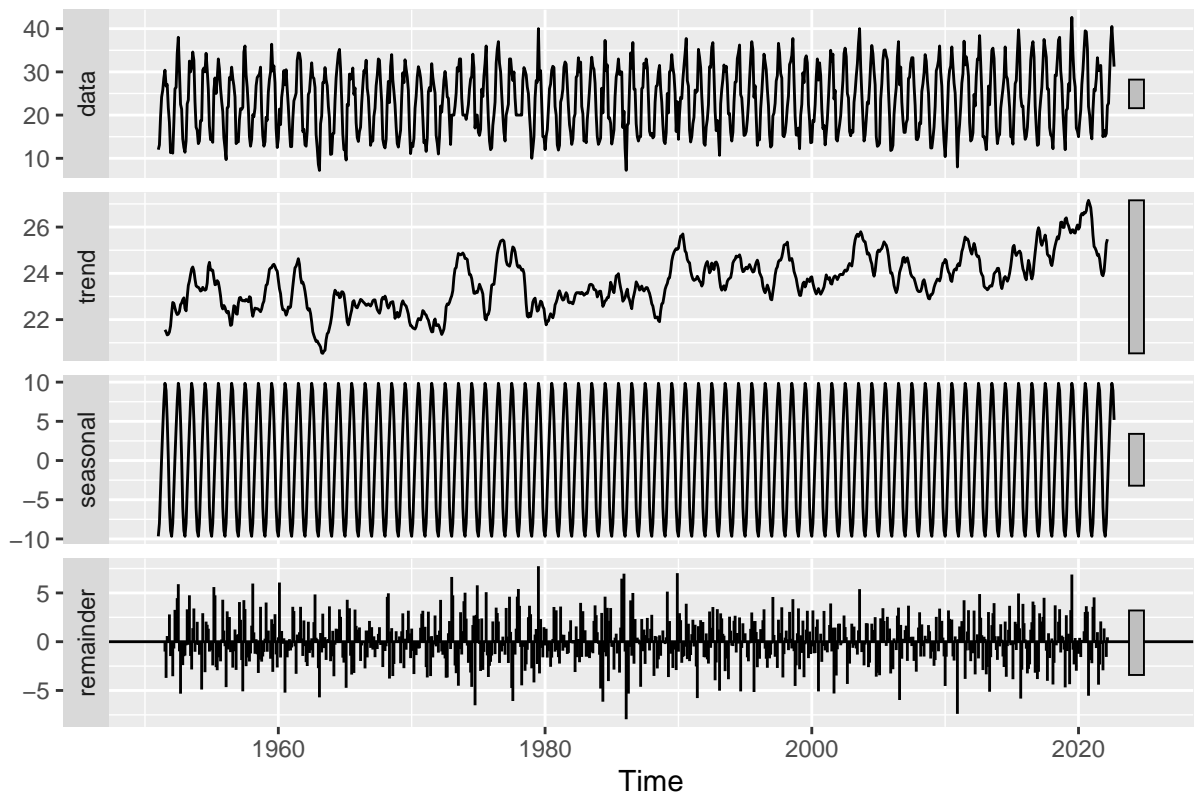
autoplot(Decomp_Temperatures_TS_Berlin)
```


Decomposition of additive time series



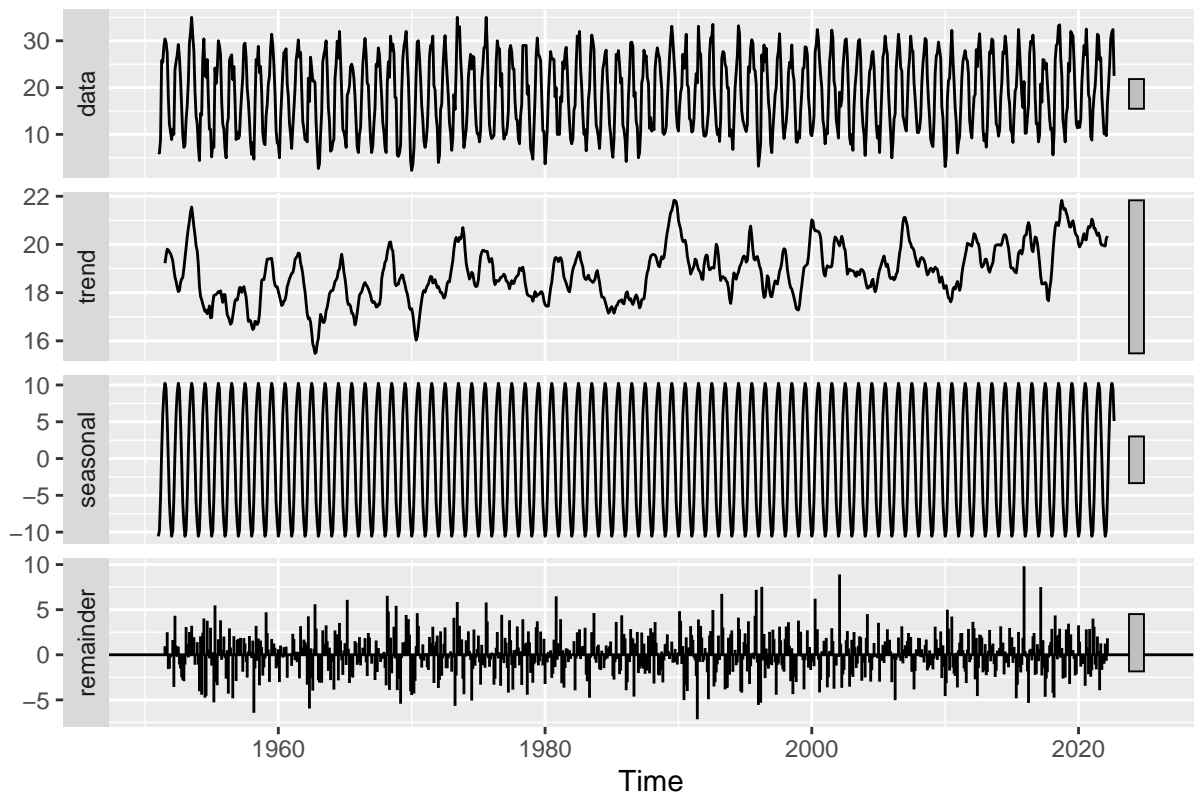
```
autoplot(Decomp_Temperatures_TS_Paris)
```

Decomposition of additive time series



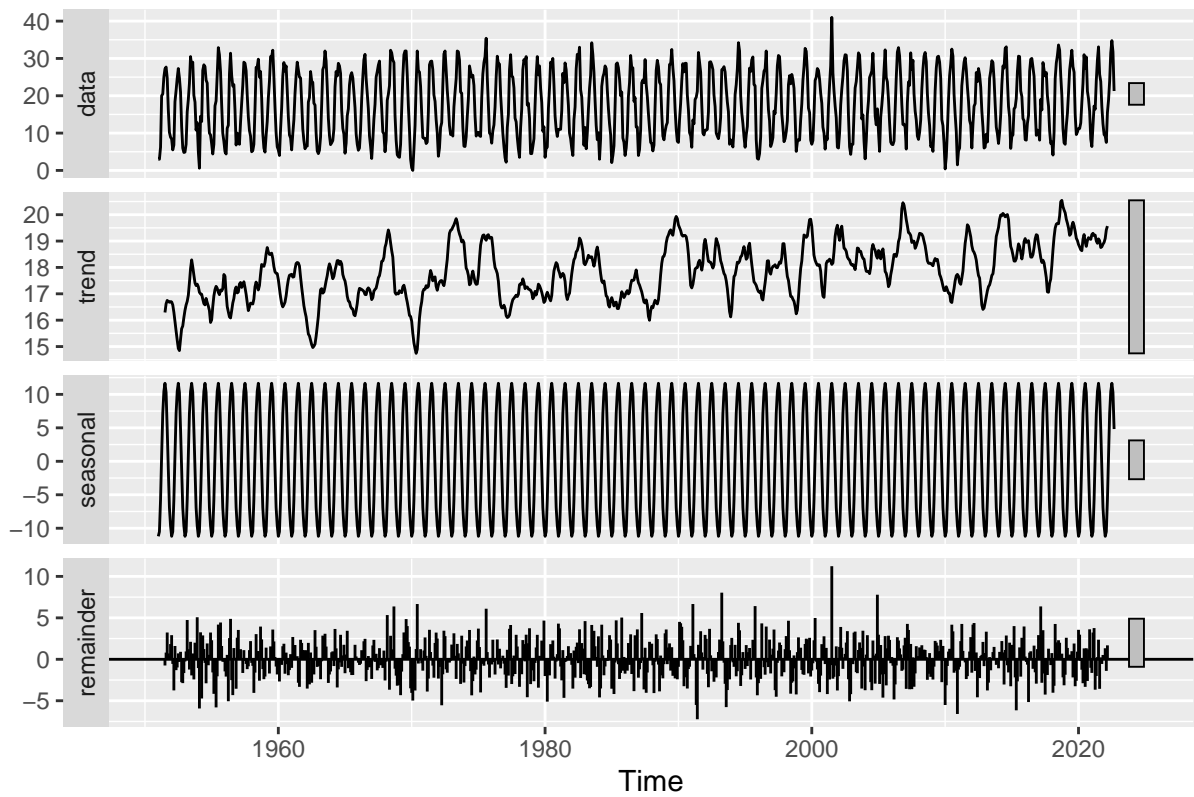
```
autoplot(Decomp_Temperatures_TS_Copenhagen)
```

Decomposition of additive time series



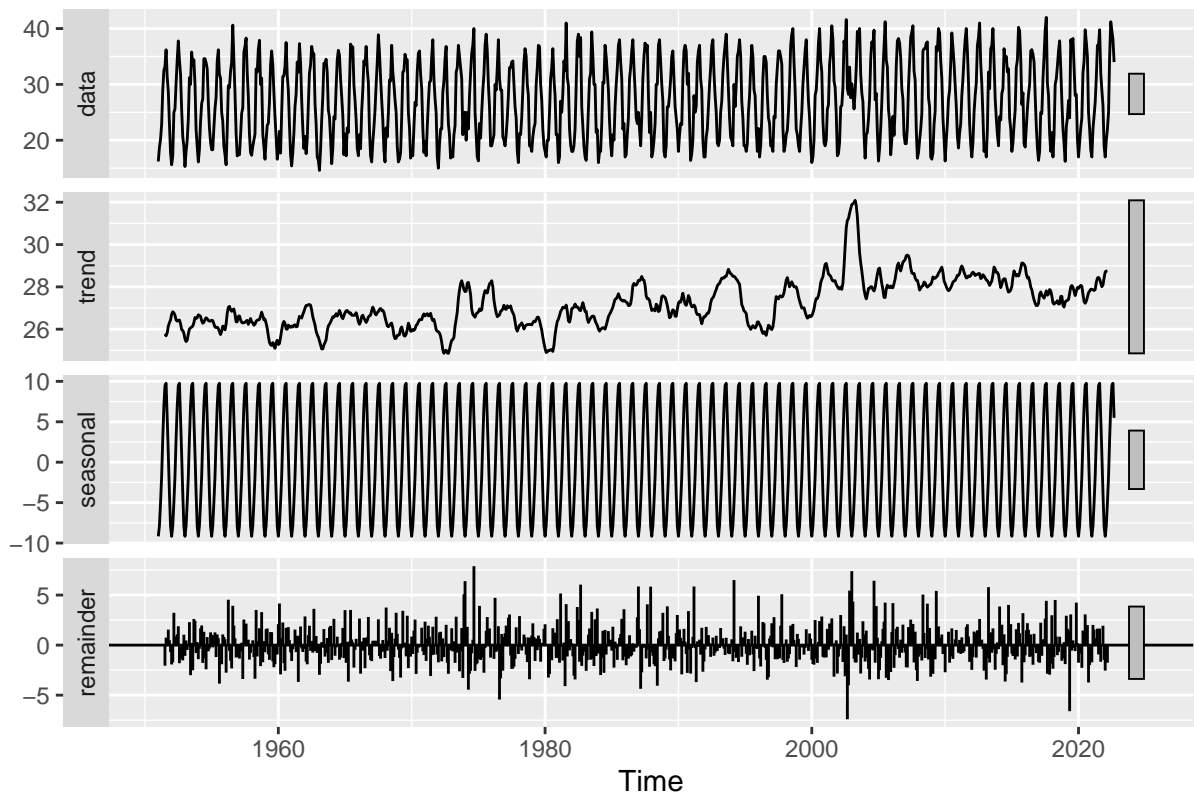
```
autoplot(Decomp_Temperatures_TS_Stockholm)
```

Decomposition of additive time series



```
autoplot(Decomp_Temperatures_TS_Rome)
```

Decomposition of additive time series



```
Box.test(Temperatures_TS[,1], type = c("Box-Pierce"), fitdf = 0)
```

```
##
## Box-Pierce test
##
## data: Temperatures_TS[, 1]
## X-squared = 523.75, df = 1, p-value < 2.2e-16
```

```
Box.test(Temperatures_TS[,1], type = c("Ljung-Box"), fitdf = 0)
```

```
##
## Box-Ljung test
##
## data: Temperatures_TS[, 1]
## X-squared = 525.58, df = 1, p-value < 2.2e-16
```

```
fit.auto <- auto.arima(Temperatures_TS[,1], seasonal = TRUE, trace = TRUE)
```

```
##
## Fitting models using approximations to speed things up...
##
## ARIMA(2,0,2)(1,1,1)[12] with drift : 4300.446
## ARIMA(0,0,0)(0,1,0)[12] with drift : 4781.884
## ARIMA(1,0,0)(1,1,0)[12] with drift : 4526.764
## ARIMA(0,0,1)(0,1,1)[12] with drift : Inf
## ARIMA(0,0,0)(0,1,0)[12] : 4779.913
## ARIMA(2,0,2)(0,1,1)[12] with drift : Inf
## ARIMA(2,0,2)(1,1,0)[12] with drift : 4525.586
## ARIMA(2,0,2)(2,1,1)[12] with drift : Inf
## ARIMA(2,0,2)(1,1,2)[12] with drift : Inf
## ARIMA(2,0,2)(0,1,0)[12] with drift : Inf
## ARIMA(2,0,2)(0,1,2)[12] with drift : Inf
## ARIMA(2,0,2)(2,1,0)[12] with drift : 4435.207
## ARIMA(2,0,2)(2,1,2)[12] with drift : Inf
## ARIMA(1,0,2)(1,1,1)[12] with drift : Inf
## ARIMA(2,0,1)(1,1,1)[12] with drift : 4298.5
## ARIMA(2,0,1)(0,1,1)[12] with drift : Inf
## ARIMA(2,0,1)(1,1,0)[12] with drift : 4525.525
## ARIMA(2,0,1)(2,1,1)[12] with drift : Inf
## ARIMA(2,0,1)(1,1,2)[12] with drift : Inf
## ARIMA(2,0,1)(0,1,0)[12] with drift : 4778.582
## ARIMA(2,0,1)(0,1,2)[12] with drift : Inf
## ARIMA(2,0,1)(2,1,0)[12] with drift : 4432.844
## ARIMA(2,0,1)(2,1,2)[12] with drift : Inf
## ARIMA(1,0,1)(1,1,1)[12] with drift : 4289.263
## ARIMA(1,0,1)(0,1,1)[12] with drift : Inf
## ARIMA(1,0,1)(1,1,0)[12] with drift : 4528.602
## ARIMA(1,0,1)(2,1,1)[12] with drift : Inf
## ARIMA(1,0,1)(1,1,2)[12] with drift : Inf
```

```

## ARIMA(1,0,1)(0,1,0)[12] with drift : 4776.299
## ARIMA(1,0,1)(0,1,2)[12] with drift : Inf
## ARIMA(1,0,1)(2,1,0)[12] with drift : 4433.311
## ARIMA(1,0,1)(2,1,2)[12] with drift : Inf
## ARIMA(0,0,1)(1,1,1)[12] with drift : Inf
## ARIMA(1,0,0)(1,1,1)[12] with drift : 4287.821
## ARIMA(1,0,0)(0,1,1)[12] with drift : Inf
## ARIMA(1,0,0)(2,1,1)[12] with drift : 4283.094
## ARIMA(1,0,0)(2,1,0)[12] with drift : 4431.334
## ARIMA(1,0,0)(2,1,2)[12] with drift : Inf
## ARIMA(1,0,0)(1,1,2)[12] with drift : Inf
## ARIMA(0,0,0)(2,1,1)[12] with drift : 4294.406
## ARIMA(2,0,0)(2,1,1)[12] with drift : 4277.927
## ARIMA(2,0,0)(1,1,1)[12] with drift : 4298.618
## ARIMA(2,0,0)(2,1,0)[12] with drift : 4432.557
## ARIMA(2,0,0)(2,1,2)[12] with drift : Inf
## ARIMA(2,0,0)(1,1,0)[12] with drift : 4528.485
## ARIMA(2,0,0)(1,1,2)[12] with drift : Inf
## ARIMA(3,0,0)(2,1,1)[12] with drift : 4277.281
## ARIMA(3,0,0)(1,1,1)[12] with drift : 4298.679
## ARIMA(3,0,0)(2,1,0)[12] with drift : 4435.082
## ARIMA(3,0,0)(2,1,2)[12] with drift : Inf
## ARIMA(3,0,0)(1,1,0)[12] with drift : 4526.9
## ARIMA(3,0,0)(1,1,2)[12] with drift : Inf
## ARIMA(4,0,0)(2,1,1)[12] with drift : 4283.697
## ARIMA(3,0,1)(2,1,1)[12] with drift : Inf
## ARIMA(4,0,1)(2,1,1)[12] with drift : 4279.952
## ARIMA(3,0,0)(2,1,1)[12] : 4277.496
##
## Now re-fitting the best model(s) without approximations...
##
## ARIMA(3,0,0)(2,1,1)[12] with drift : Inf
## ARIMA(3,0,0)(2,1,1)[12] : Inf

```

```

## ARIMA(2,0,0)(2,1,1)[12] with drift      : Inf
## ARIMA(4,0,1)(2,1,1)[12] with drift      : Inf
## ARIMA(1,0,0)(2,1,1)[12] with drift      : Inf
## ARIMA(4,0,0)(2,1,1)[12] with drift      : Inf
## ARIMA(1,0,0)(1,1,1)[12] with drift      : Inf
## ARIMA(1,0,1)(1,1,1)[12] with drift      : Inf
## ARIMA(0,0,0)(2,1,1)[12] with drift      : Inf
## ARIMA(2,0,1)(1,1,1)[12] with drift      : Inf
## ARIMA(2,0,0)(1,1,1)[12] with drift      : Inf
## ARIMA(3,0,0)(1,1,1)[12] with drift      : Inf
## ARIMA(2,0,2)(1,1,1)[12] with drift      : Inf
## ARIMA(1,0,0)(2,1,0)[12] with drift      : 4469.039
##
## Best model: ARIMA(1,0,0)(2,1,0)[12] with drift

```