Assignment-1

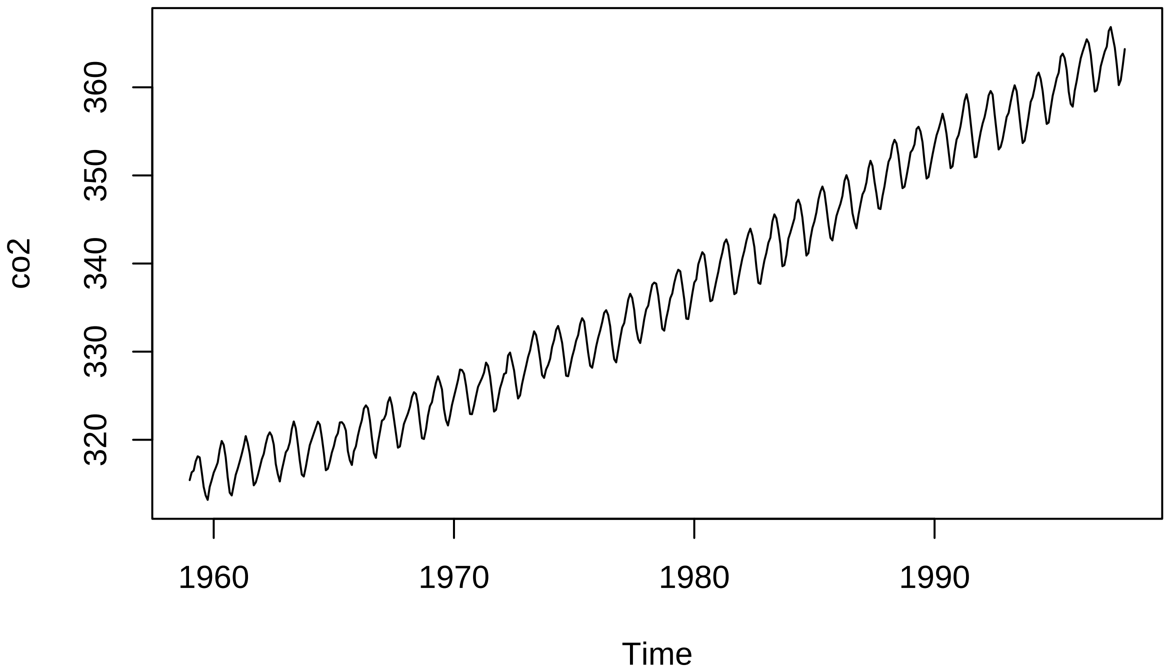
2023-03-01

Exercise 1

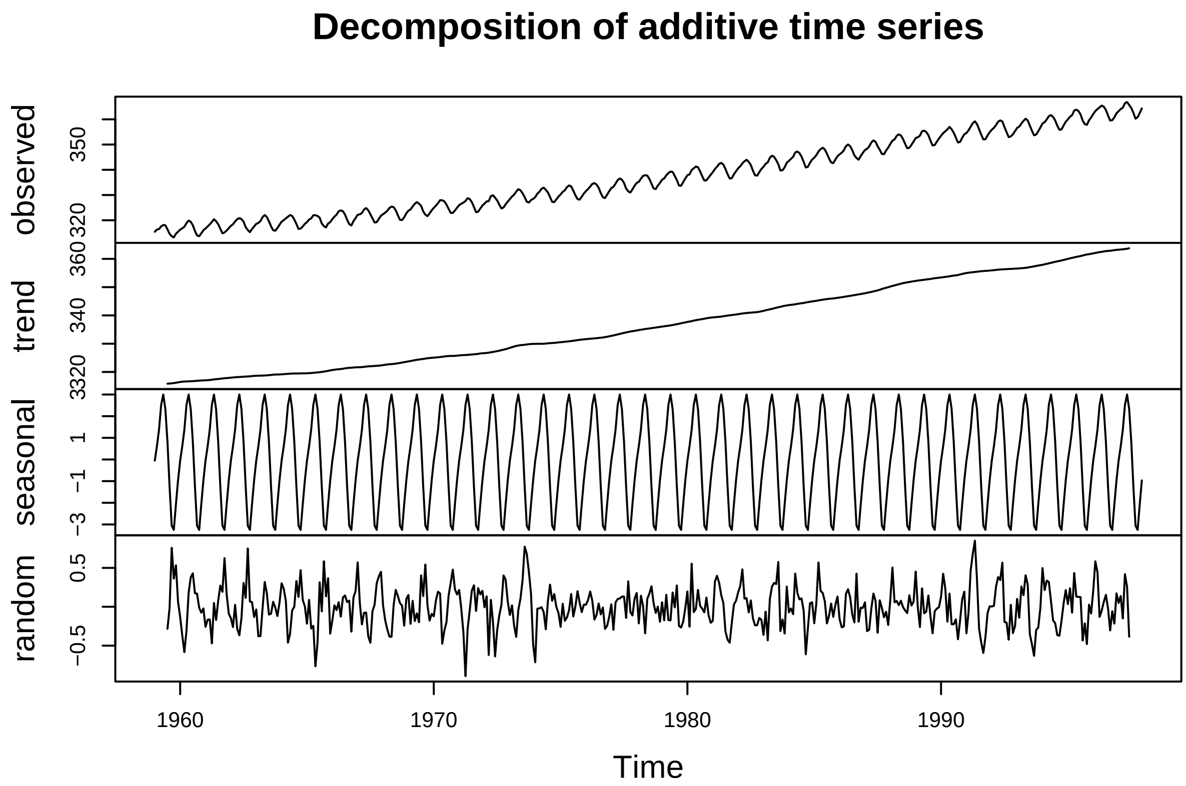
```r
str(co2)
```

```
##  Time-Series [1:468] from 1959 to 1998: 315 316 316 318 318 ...
```

```r
plot(co2)
```




```r
co2_d <- decompose(co2, type = "additive")
plot(co2_d)
```



Exercise 2

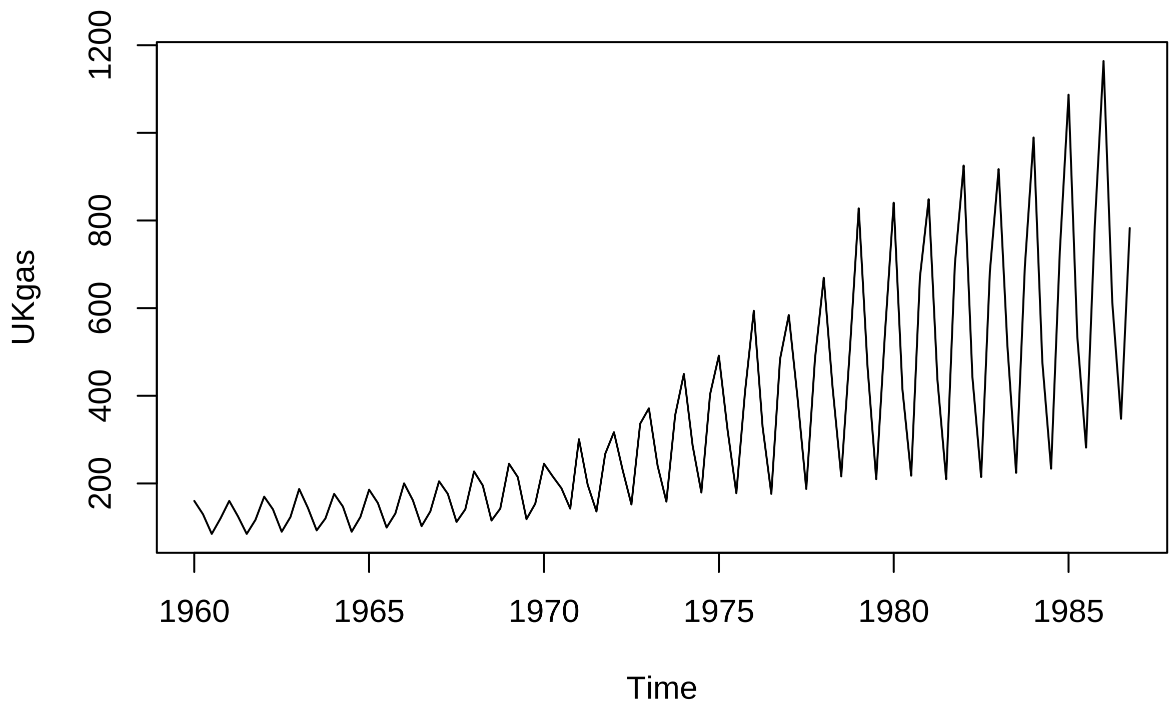
```
str(UKgas)
```

```
##  Time-Series [1:108] from 1960 to 1987: 160.1 129.7 84.8 120.1 160.1 ...
```

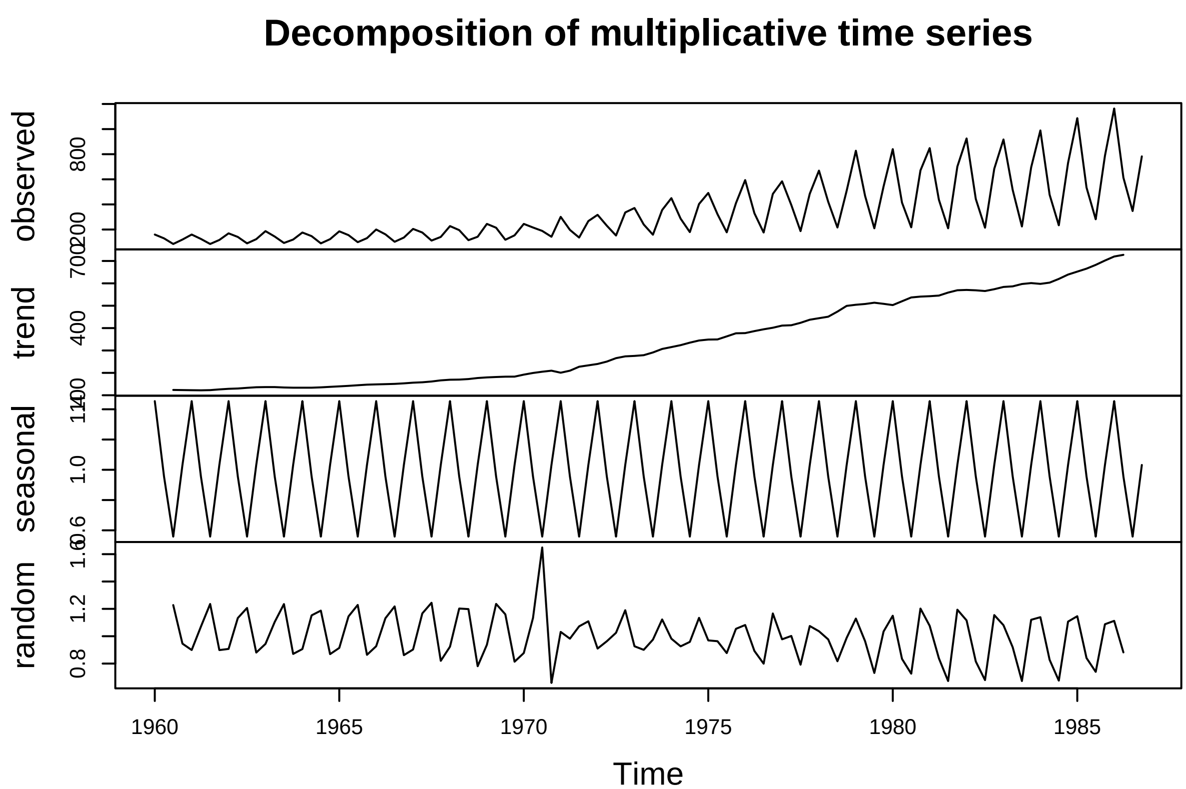
```
library(fpp2)
```

```
## Warning: package 'fpp2' was built under R version 4.1.2

## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo

## -- Attaching packages ---------------------------------------------- fpp2 2.5 --

## v ggplot2   3.3.5     v fma       2.5
## v forecast  8.21      v expsmooth 2.3

## Warning: package 'forecast' was built under R version 4.1.2

## Warning: package 'fma' was built under R version 4.1.2

##
```

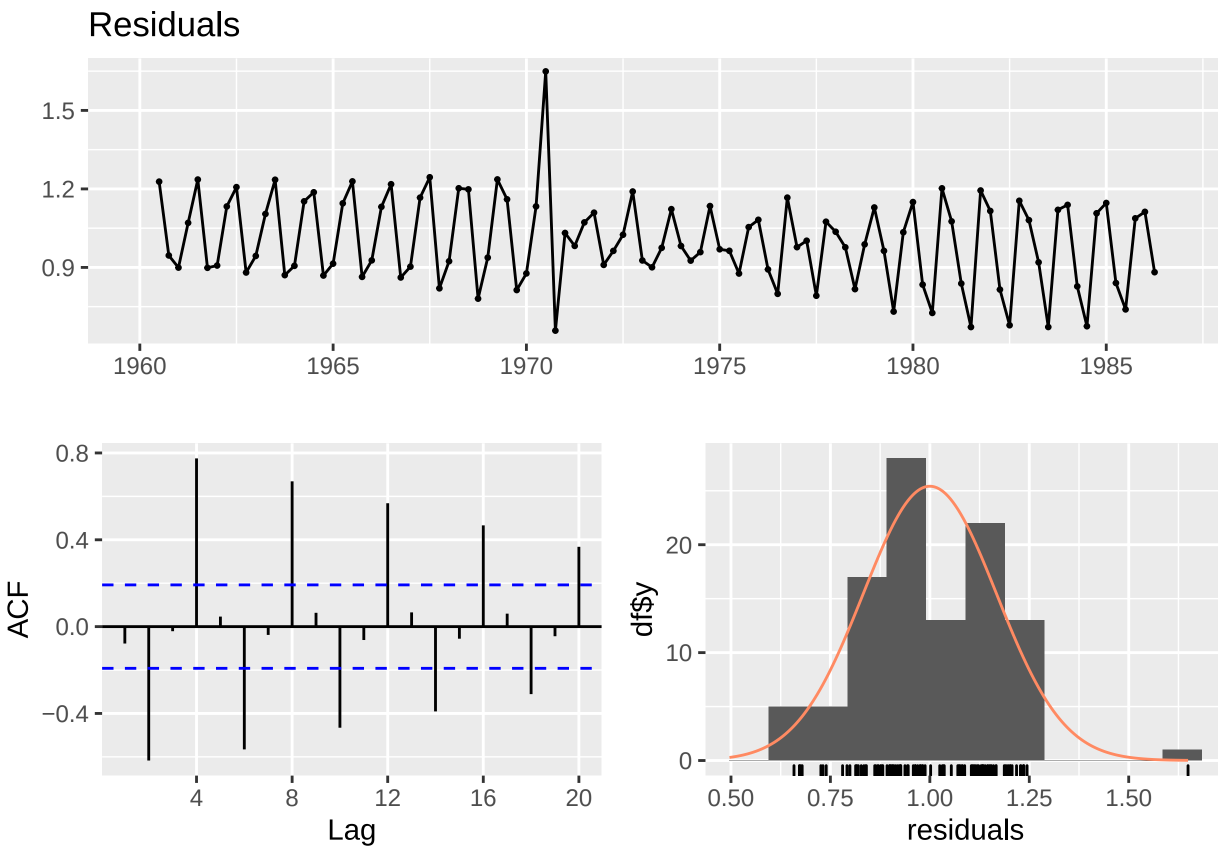
```
library(forecast)

plot(UKgas)
```




```r
UKgas_d <- decompose(UKgas, type = "multiplicative")
plot(UKgas_d)
```

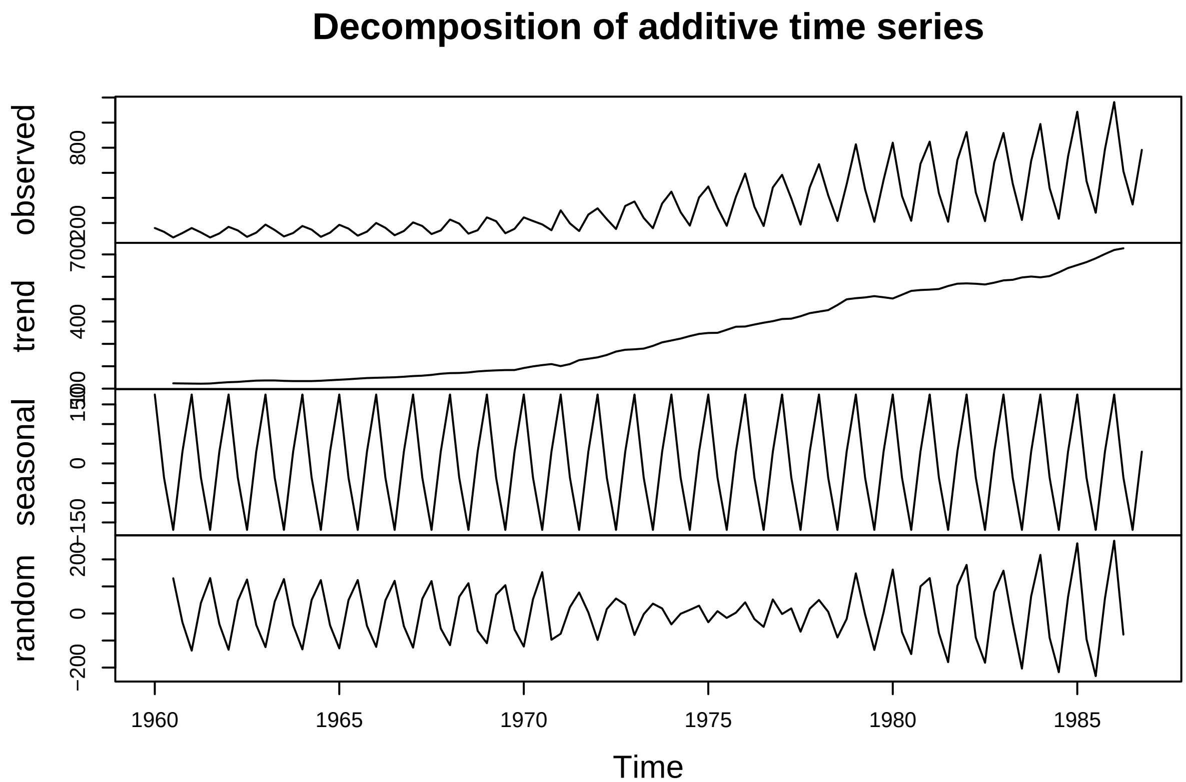



```r
checkresiduals(remainder(UKgas_d))
```

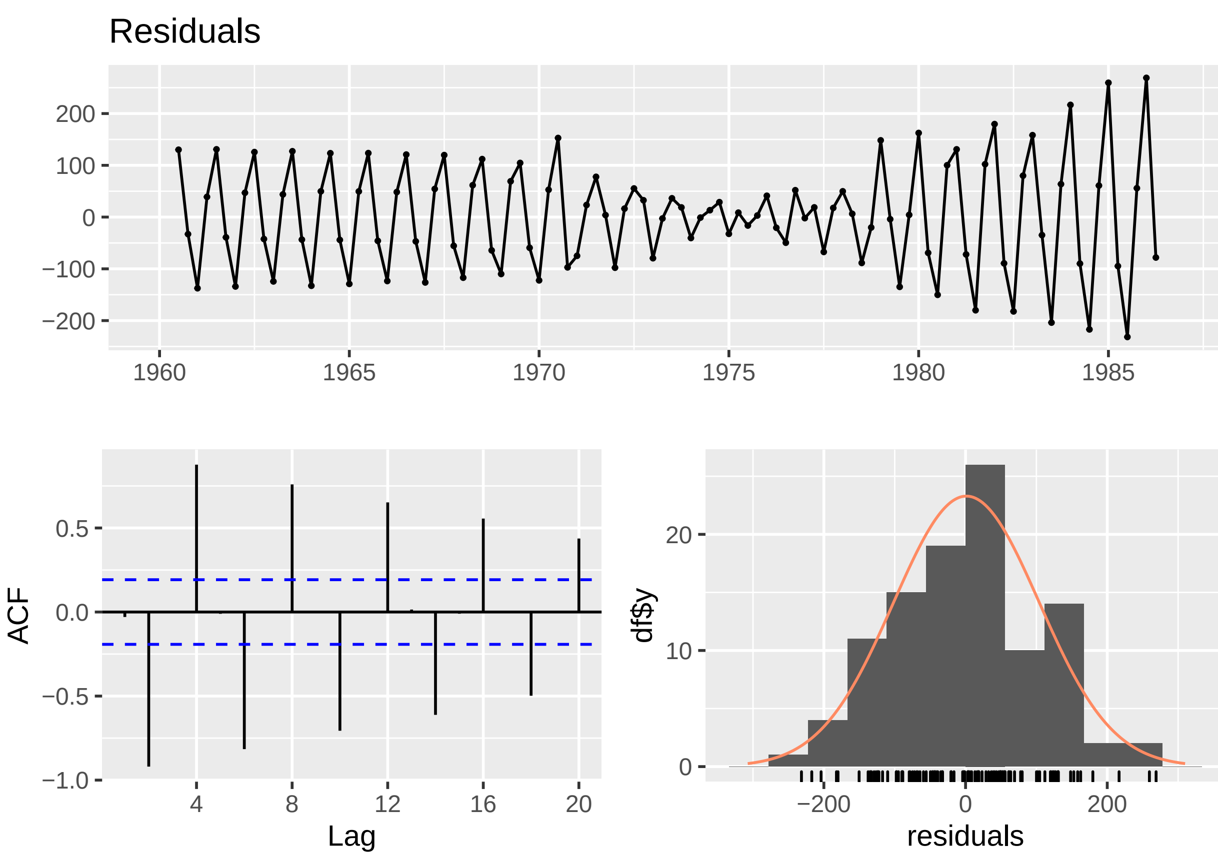



```
##
##  Ljung-Box test
##
## data:  Residuals
## Q* = 195.83, df = 8, p-value < 2.2e-16
##
## Model df: 0.   Total lags used: 8
```

```
UKgas_d2 <- decompose(UKgas, type = "additive")
plot(UKgas_d2)
```




```
checkresiduals(remainder(UKgas_d2))
```

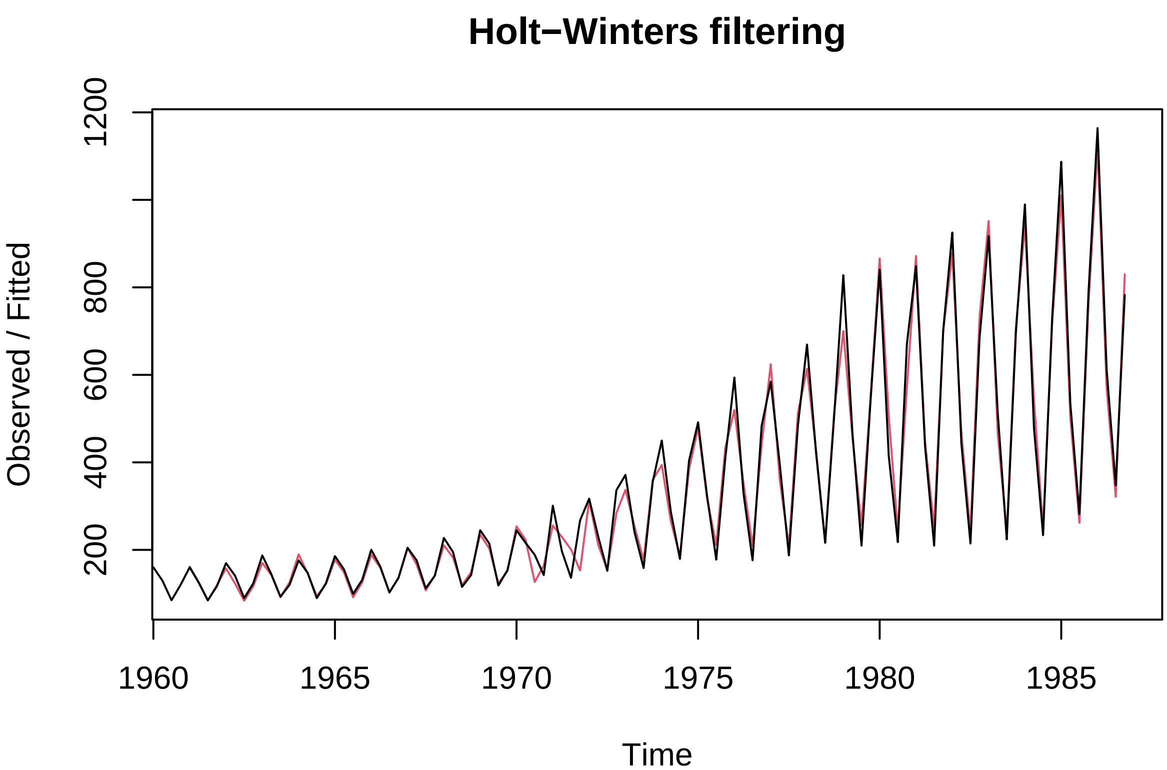


```
##
##  Ljung-Box test
##
## data:  Residuals
## Q* = 317.25, df = 8, p-value < 2.2e-16
##
## Model df: 0.   Total lags used: 8
```

As we can see from the plotted data, there is a change in the seasonal component from the beginning of the 70's. At the same time there is also a change in the trend, which becomes steeper around the same year. In this case, a multiplicative decomposition is the more appropriate approach, since we do not have a constant seasonal component. What can be done alternatively is that the data is first transformed to be stable over time before the additive decomposition is used. In this case log transformation could be used. The reason for those methods is that $y_t = S_t \times T_t \times R_t$ is the same as $log(y_t) = log(S_t) + log(T_t) + log(R_t)$

Multiplicative decomposition being more appropriate can be shown by analyzing the residuals of the two decompositions: we can see that in the case of the multiplicative time series the residuals are smaller, suggesting a better fit for the raw data.

```r
HWGas <- HoltWinters(UKgas)
plot(HWGas)
```




```r
HWGas
```

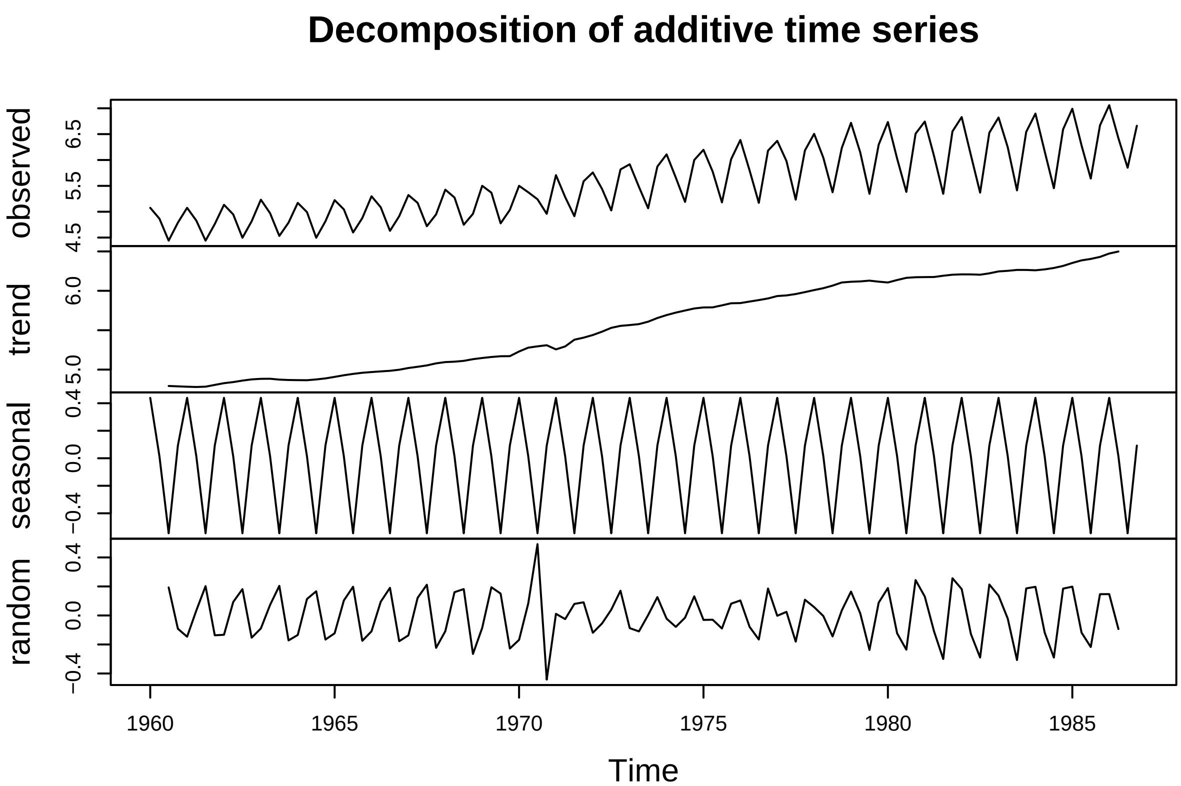
```
## Holt-Winters exponential smoothing with trend and additive seasonal component.
##
## Call:
## HoltWinters(x = UKgas)
##
## Smoothing parameters:
##  alpha: 0.02127689
##  beta : 1
```

```
##   gamma: 0.9899092
##
## Coefficients:
##           [,1]
## a   621.06125
## b    10.12142
## s1  574.20494
## s2   12.86129
## s3 -263.79964
## s4  162.20290
```

Add comments here on a comparison of those results to previous ones.

Beta is equal to 1 meaning the trend slope seems to be very dependent on the previous trend slope. Gamma is very close to 1 too meaning that more weighing is given to the most recent seasonal cycles. Alpha being very close to 0.

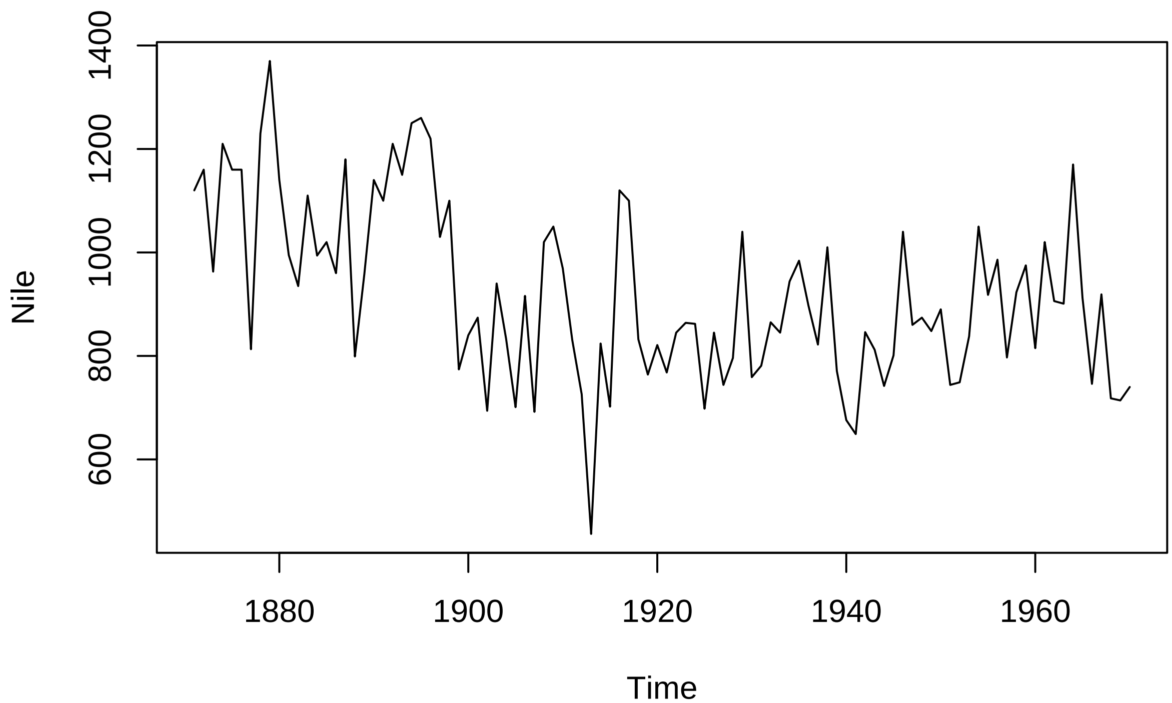
```
decUK <- decompose(log(UKgas),type = "additive")
plot(decUK)
```



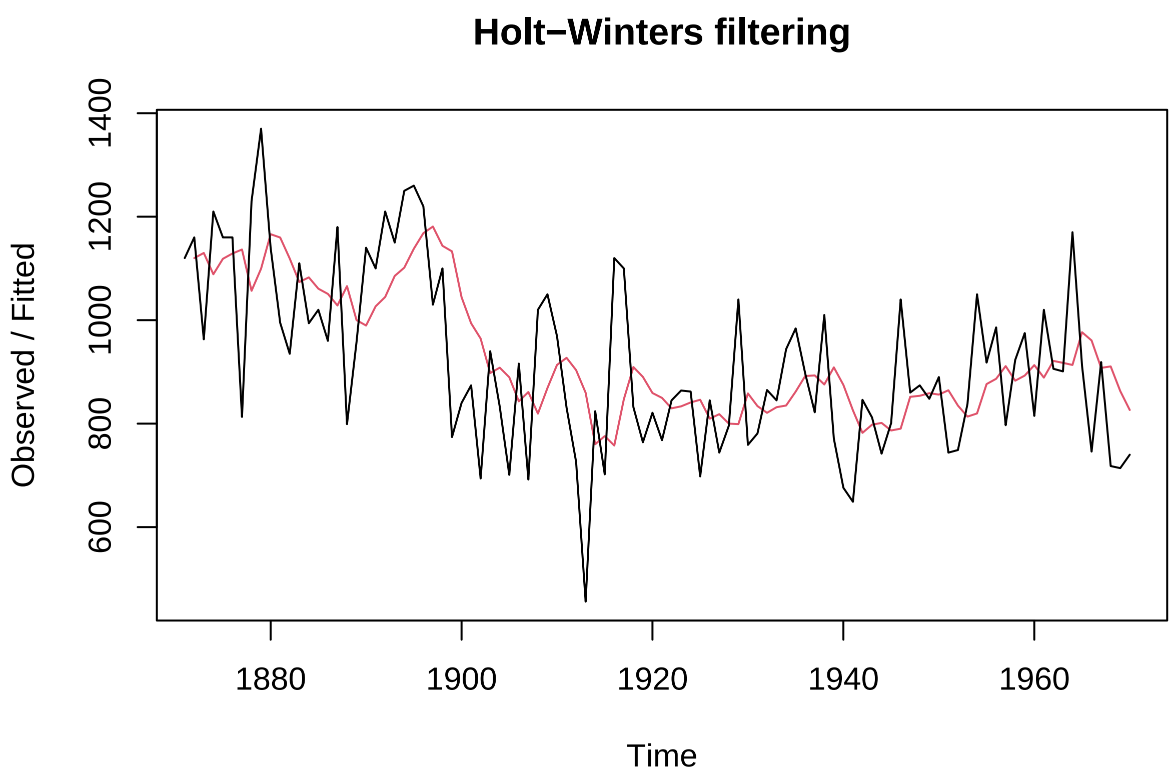
##

Exercise 3

```
plot(Nile)
```




```
HWNile<- HoltWinters(Nile,  beta= F, gamma= F)
plot(HWNile)
```

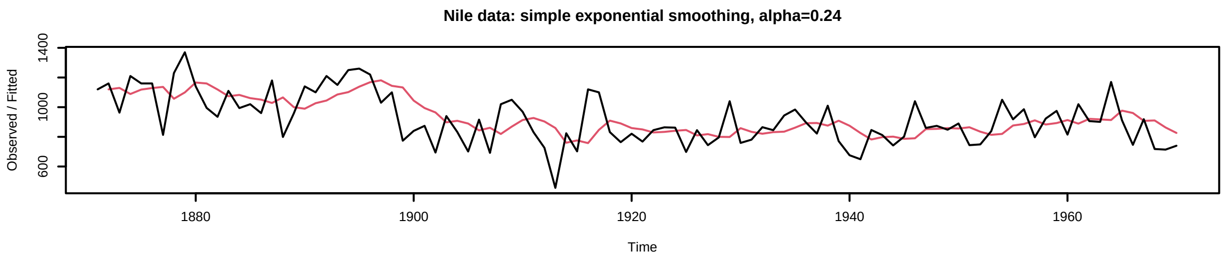


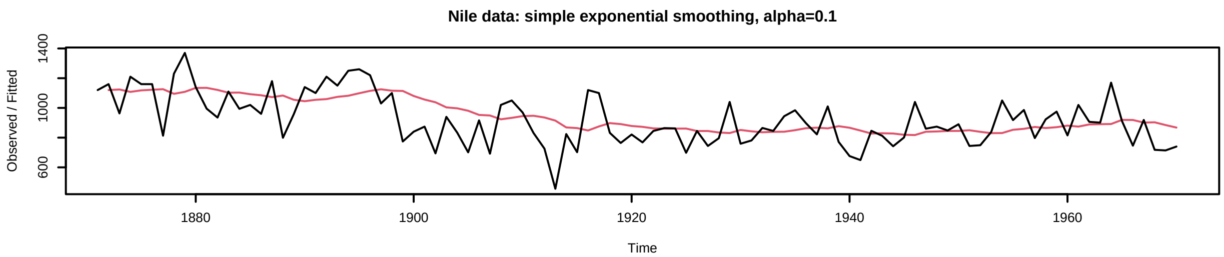

```
HWNile
```

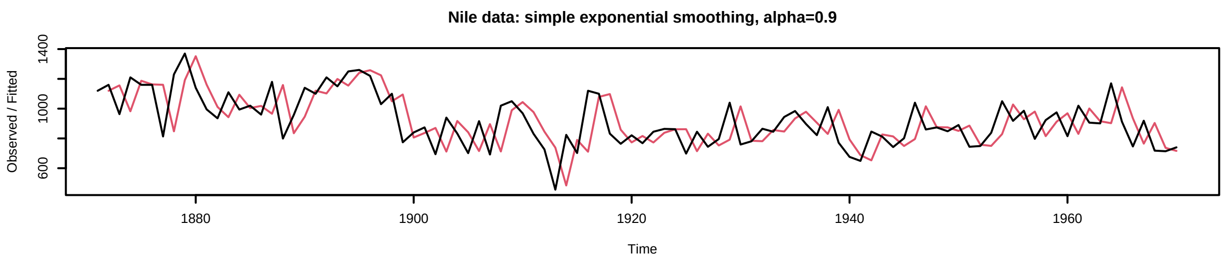
```
## Holt-Winters exponential smoothing without trend and without seasonal component.
##
## Call:
```

```
## HoltWinters(x = Nile, beta = F, gamma = F)
##
## Smoothing parameters:
##  alpha: 0.2465579
##  beta : FALSE
##  gamma: FALSE
##
## Coefficients:
##       [,1]
## a 805.0389
```

```r
par(mfrow=c(3,1), cex=.4)
plot(HWNile, main="Nile data: simple exponential smoothing, alpha=0.24")

HWNile2 <- HoltWinters(Nile, alpha=.1, beta=F, gamma=F)
plot(HWNile2, main="Nile data: simple exponential smoothing, alpha=0.1")

HWNile3 <- HoltWinters(Nile, alpha=.9, beta=F, gamma=F)
plot(HWNile3, main="Nile data: simple exponential smoothing, alpha=0.9")
```







Add comments here on the effect of Alpha on the forecasts Smaller values of α show smoother forecast function while α closer to 1 shows a function that is not as smooth and follows the original function quite closely. The default α is chosen after tuning-in the parameters and minimizing the errors between the predicted values and the actual ones.

Exercise 4

read.table("coronavirus-data.txt")