Homework2_201611531

Jeong Hojae

2021 3 27

0. Read the data

```
library(readr)
library(ggplot2)
library(forecast)
```

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

```
library(tseries)

spot <- read_csv("/cloud/project/spot.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   Spot = col_double()
## )
```

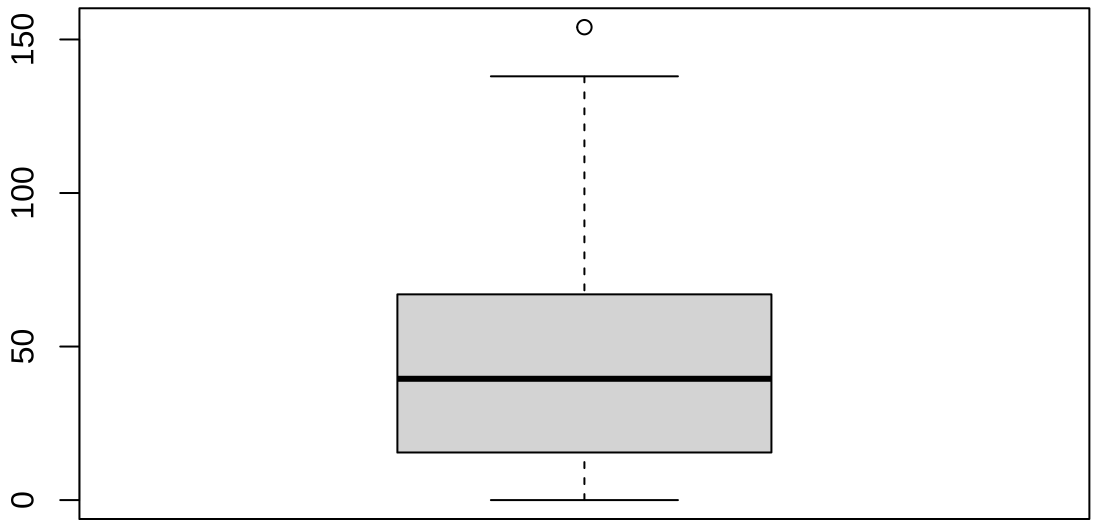
```
dim(spot)
```

```
## [1] 100   1
```

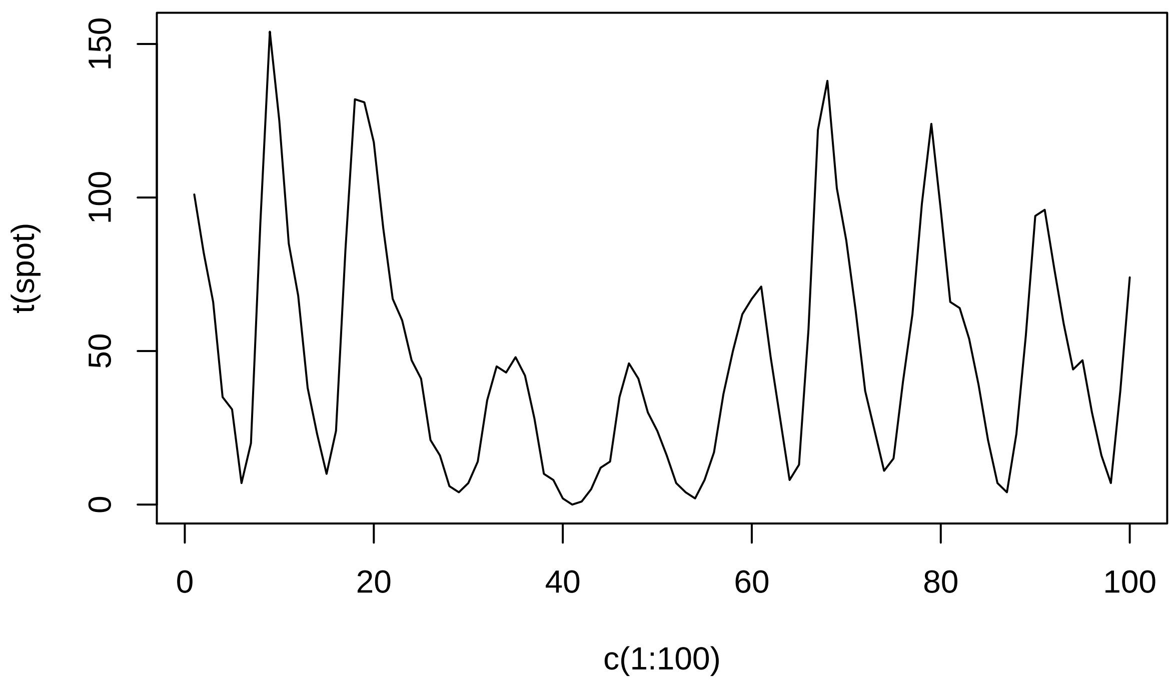
```
summary(spot)
```

```
##       Spot
##  Min.   :  0.00
##  1st Qu.: 15.75
##  Median : 39.50
##  Mean   : 46.93
##  3rd Qu.: 67.00
##  Max.   :154.00
```

```
boxplot(spot)
```

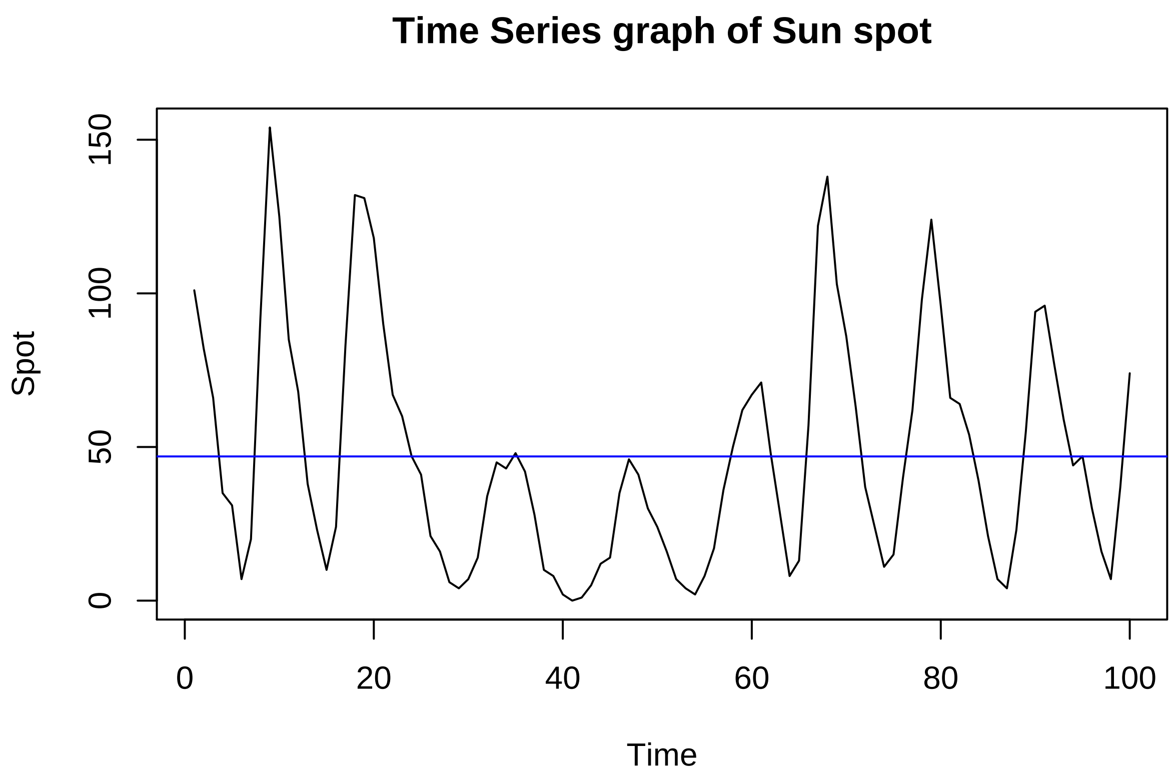



```
plot(c(1:100), t(spot),type = 'l')
```



1. Draw time series graph of sun spot. Do you see any cycle or seasonal effect?

```
spot.ts=ts(data=spot, frequency=1)
plot(spot.ts, mai ="Time Series graph of Sun spot")
abline(h=mean(spot.ts[,1]),col="blue")
```




```r
tseries::kpss.test(spot.ts,null="Level")
```

```
## Warning in tseries::kpss.test(spot.ts, null = "Level"): p-value greater than
## printed p-value
```

```
##
##  KPSS Test for Level Stationarity
##
## data:  spot.ts
## KPSS Level = 0.15966, Truncation lag parameter = 4, p-value = 0.1
```

```r
tseries::kpss.test(spot.ts,null="Trend")
```

```
##
##  KPSS Test for Trend Stationarity
##
## data:  spot.ts
## KPSS Trend = 0.15776, Truncation lag parameter = 4, p-value = 0.0402
```

H0 : Data is level stationary. (tseries::kpss.test(spot.ts, null="Level"))

When significance level is 0.05, p-value (0.1) is greater than it. So, We cannot reject H0.

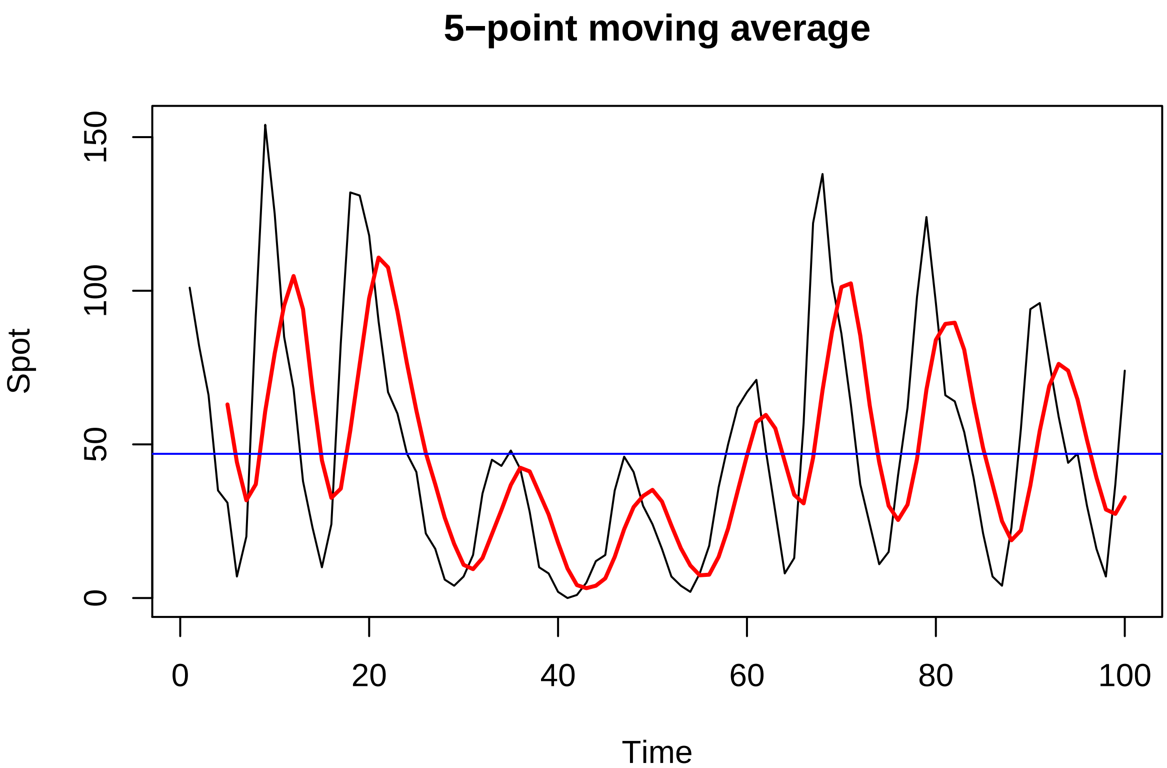
H0 : Data is trend stationary. (tseries::kpss.test(spot.ts, null="Trend"))

When significance level is 0.05, p-value (0.0402) is smaller than it. So, We can reject H0.

Therefore, This data has level. Alse, we can see cycle or seasonal effect easily at the graph.

2. try 5-point moving average smoothing. Draw the plot of original graph in black, 5 point MA smoothing in red, mean value in blue.

```
m5=filter(spot.ts, filter=rep(1/5,5), method="convolution", sides=1)
plot(spot.ts,main="5-point moving average")
lines(m5,col="red",lty=1, lwd=2)
abline(h=mean(spot.ts[,1]),col="blue")
```



There is no center to calculate average.

$M_5 = [Y_5 + Y_4 + Y_3 + Y_2 + Y_1]/5$

3. Check the residual plot, check the stationary and the test the independence assumption. Carefully interpret the residual analysis.

```
head(m5,10)
```

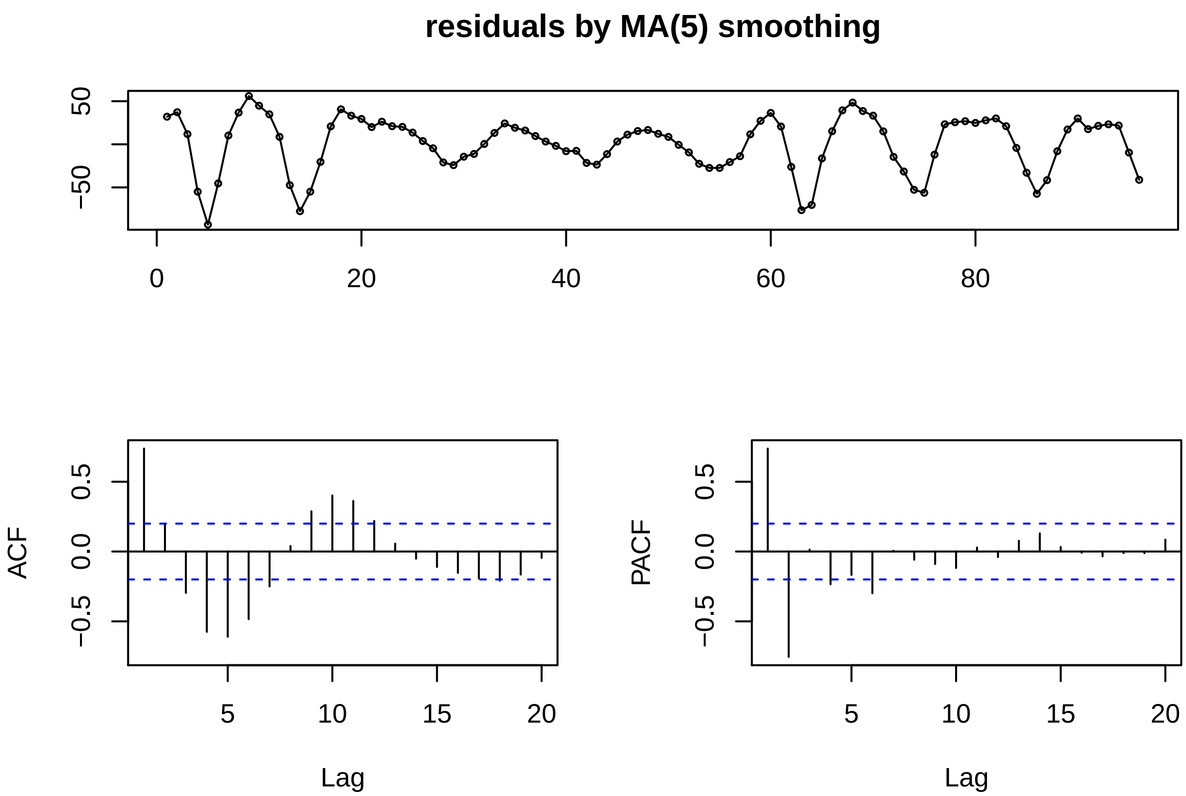
```
## Time Series:
## Start = 1
## End = 10
## Frequency = 1
##       [,1]
##  [1,]   NA
##  [2,]   NA
##  [3,]   NA
##  [4,]   NA
##  [5,] 63.0
##  [6,] 44.2
##  [7,] 31.8
##  [8,] 37.0
```

```
##  [9,] 60.8
## [10,] 79.6
```

```r
res=m5[-1:-4,]-spot.ts[-1:-4,]
head(res,10)
```

```
##  [1]  32.0  37.2  11.8 -55.0 -93.2 -45.4  10.2  36.8  56.0  44.8
```

```r
tsdisplay(res, main="residuals by MA(5) smoothing")
```



At the residual plot, it seems to have pattern.

Also, several bars are crossing the blue line (significance level) at the ACF and PACF graphs.

Therefore, The residual shows auto correlation after the model MA(5) is applied. i.e. It is non-stationary.

```r
Box.test(res)
```

```
## 
##  Box-Pierce test
## 
## data:  res
## X-squared = 52.188, df = 1, p-value = 5.044e-13
```

H0 : given time series data is independence

When significance level is 0.05, p-value (5.044e-13) is smaller than it. So, We can reject H0.

Therefore, independence of residuals doesn't exist.

4. Fit the simple exponential smoothing with alpha=0.1 and with the optimized alpha. If you think we need a trend, or seasonal, or both try them. Please address all the modeling and show how you find the best exponential smoothing model for spot data.

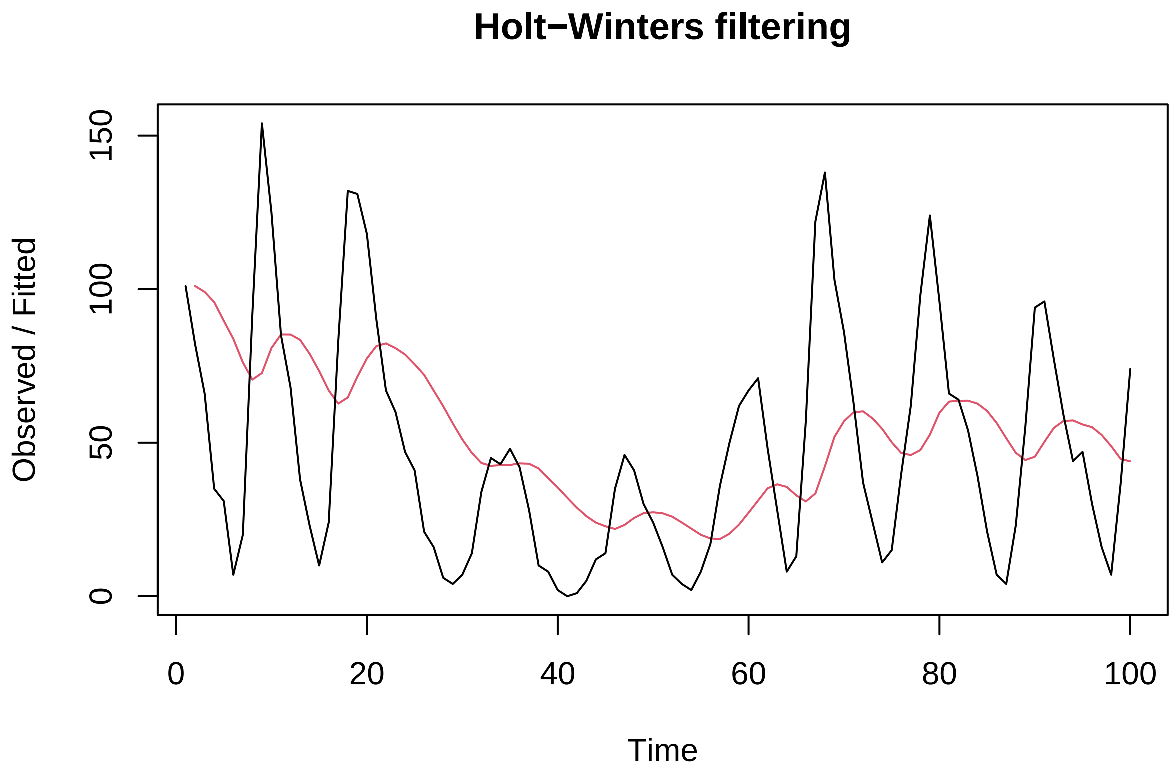
Let's compare exponential smoothing in all cases.

no trend and no seasonal effect

```
ho=HoltWinters(spot.ts, alpha=0.1, beta=F, gamma=F)
#exponential smoothing # (beta=F, gamma=F): no trend and no seasonal effect
ho
```

```
## Holt-Winters exponential smoothing without trend and without seasonal component.
##
## Call:
## HoltWinters(x = spot.ts, alpha = 0.1, beta = F, gamma = F)
##
## Smoothing parameters:
##  alpha: 0.1
##  beta : FALSE
##  gamma: FALSE
##
## Coefficients:
##       [,1]
## a 46.92938
```

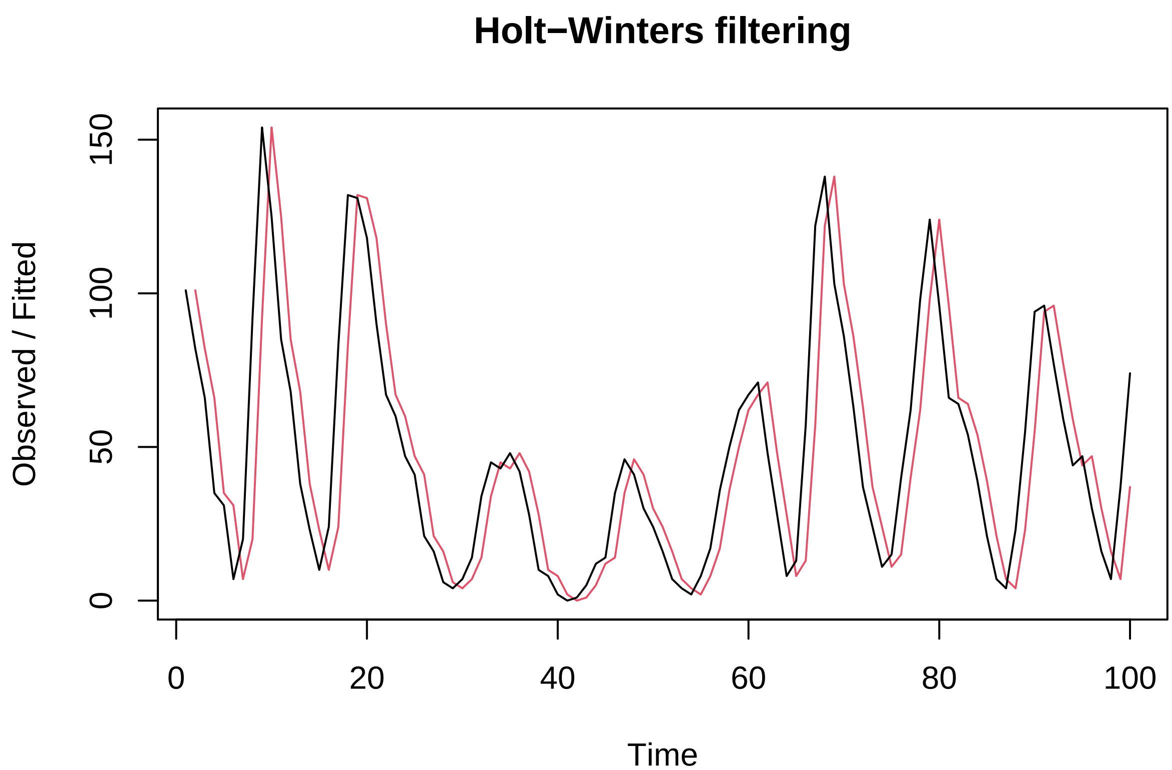
```
plot(ho)
```



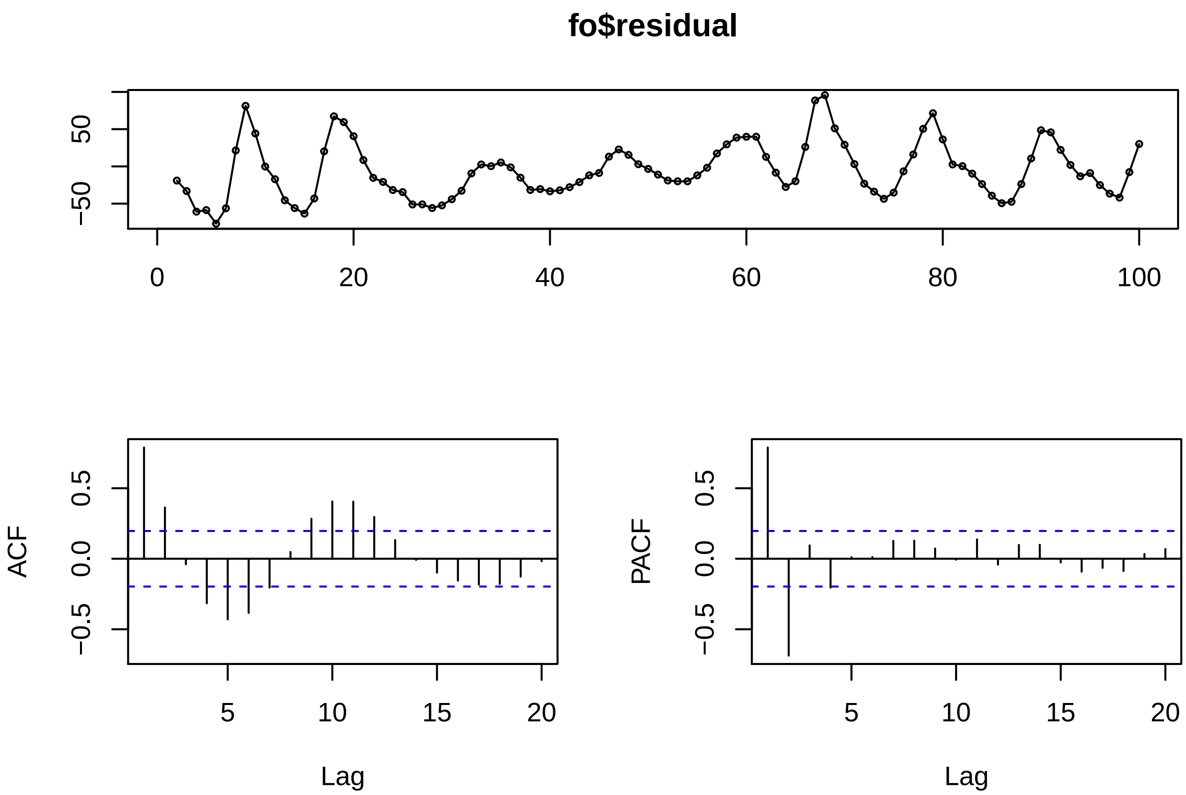

```
ha=HoltWinters(spot.ts,beta=F, gamma=F) #exponential smoothing
ha
```

```
## Holt-Winters exponential smoothing without trend and without seasonal component.
##
## Call:
## HoltWinters(x = spot.ts, beta = F, gamma = F)
##
## Smoothing parameters:
##  alpha: 0.9999339
##  beta : FALSE
##  gamma: FALSE
##
## Coefficients:
##       [,1]
## a 73.99755
```

```
plot(ha) # the red line is the fitted value
```



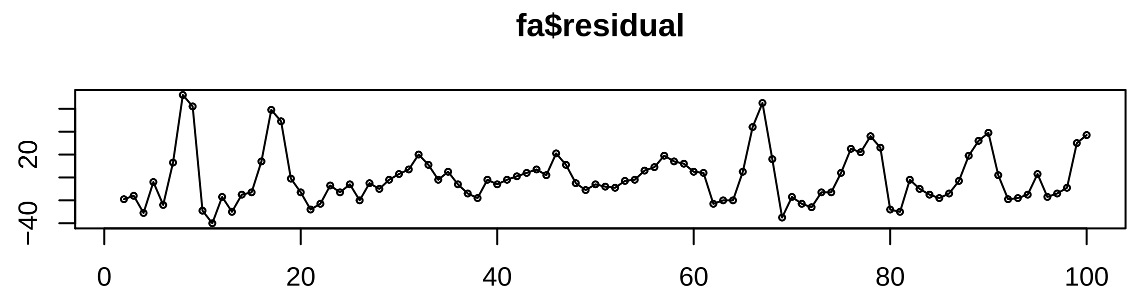

```
fo=forecast(ho)
tsdisplay(fo$residual)
```

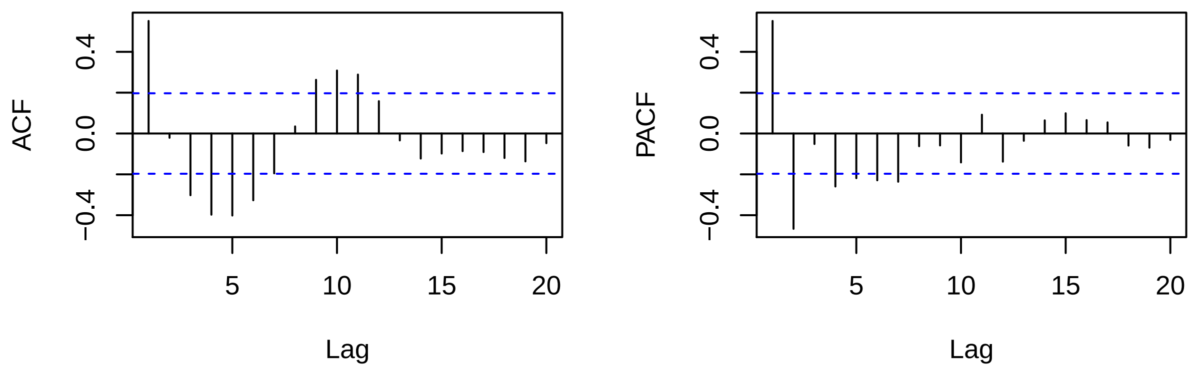



```
Box.test(fo$residual, type="Box-Pierce")
```

```
##
##  Box-Pierce test
##
## data:  fo$residual
## X-squared = 61.663, df = 1, p-value = 4.108e-15
```

```
fa=forecast(ha)
tsdisplay(fa$residual)
```






```
Box.test(fa$residual, type="Box-Pierce")
```

```
##
##  Box-Pierce test
##
## data:  fa$residual
## X-squared = 30.077, df = 1, p-value = 4.153e-08
```

```
accuracy(fo)
```

```
##                     ME     RMSE      MAE  MPE MAPE     MASE      ACF1
## Training set -5.461679 36.69865 29.97745 -Inf  Inf 1.748832 0.7892169
```

```
accuracy(fa)
```

```
##                    ME     RMSE      MAE  MPE MAPE     MASE      ACF1
## Training set -0.27277 22.45279 17.14213 -Inf  Inf 1.000042 0.5511841
```

no trend effect

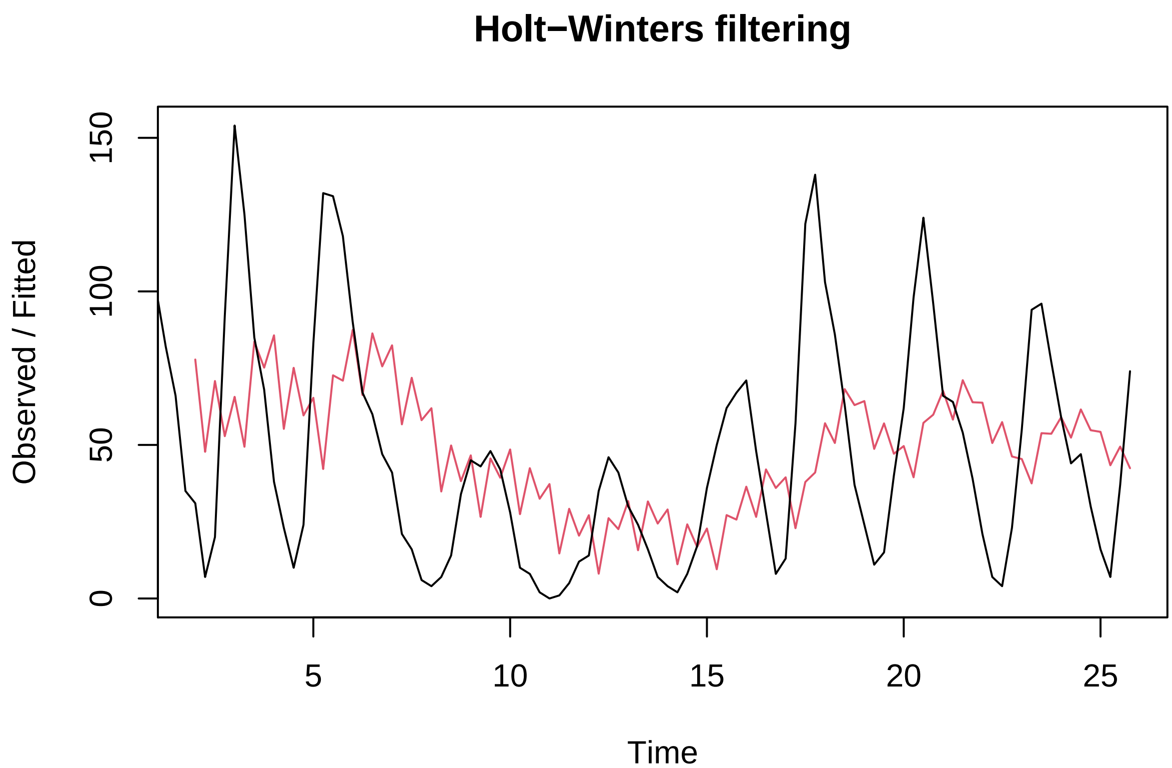
To consider seasonal variation, I set up frequency to 4.

```
spot.ts4=ts(data=spot,frequency=4) # seasonal data
ho=HoltWinters(spot.ts4, alpha=0.1, beta=F) #exponential smoothing # (beta=F): no trend effect
ho
```

```
## Holt-Winters exponential smoothing without trend and with additive seasonal component.
##
## Call:
## HoltWinters(x = spot.ts4, alpha = 0.1, beta = F)
##
## Smoothing parameters:
```

```
##  alpha: 0.1
##  beta : FALSE
##  gamma: 0.05303838
##
## Coefficients:
##         [,1]
## a  48.611698
## s1 -1.731763
## s2 -8.700668
## s3  2.169332
## s4 -1.528832
```

```
plot(ho)
```

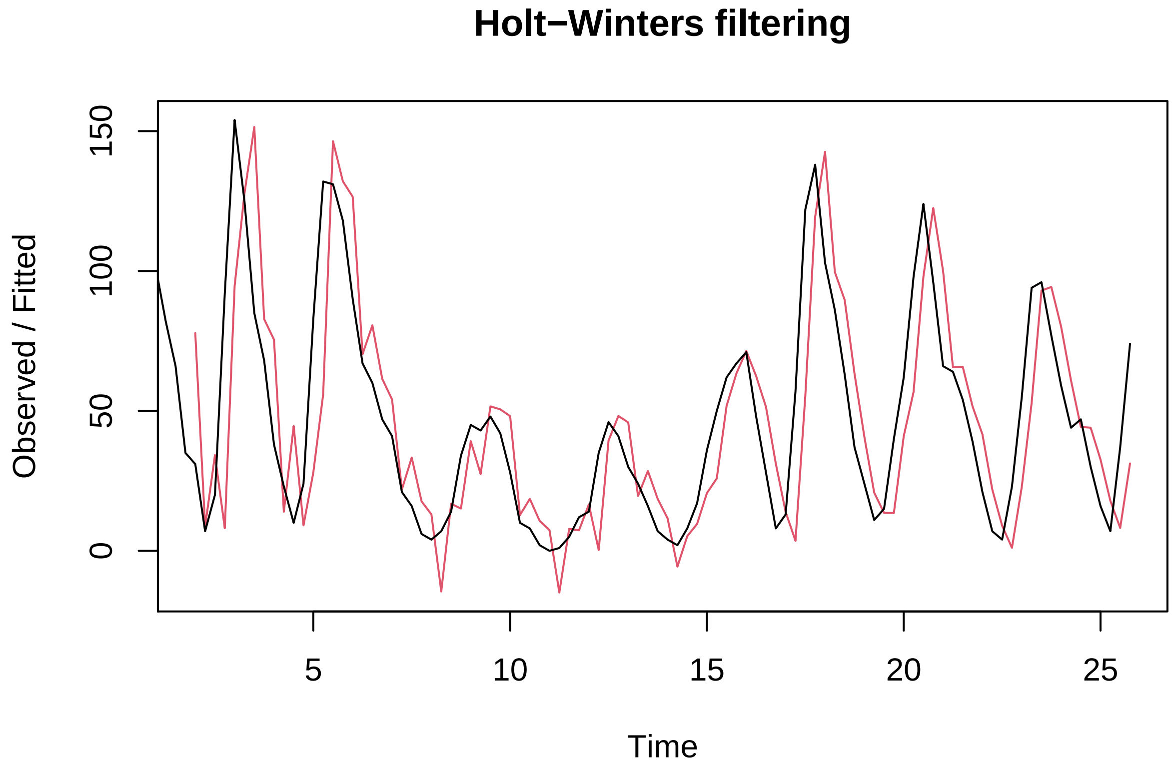



```
ha=HoltWinters(spot.ts4,beta=F) #exponential smoothing
ha
```

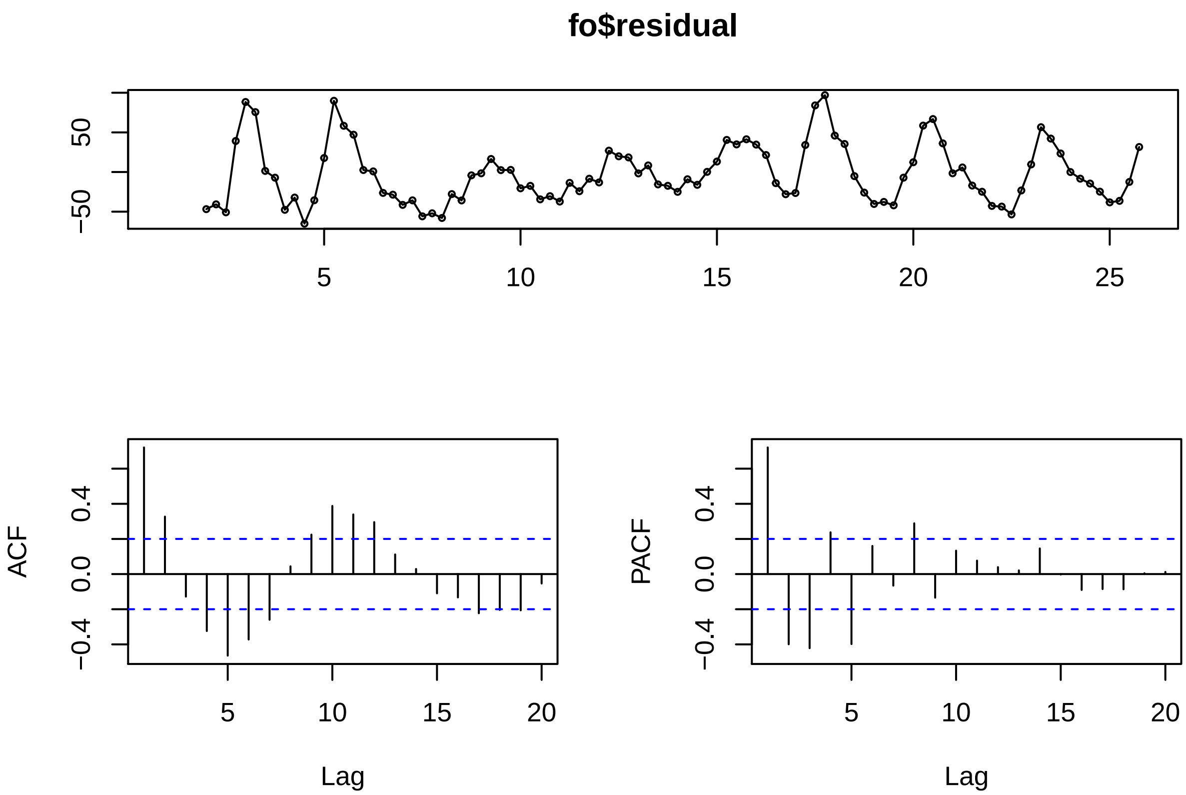
```
## Holt-Winters exponential smoothing without trend and with additive seasonal component.
##
## Call:
## HoltWinters(x = spot.ts4, beta = F)
##
## Smoothing parameters:
##  alpha: 0.9350651
##  beta : FALSE
##  gamma: 1
##
## Coefficients:
##          [,1]
## a  74.3403045
## s1 -1.5697735
```

```
## s2 -0.3844468
## s3  2.6566979
## s4 -0.3403045
```

```
plot(ha) # the red line is the fitted value
```



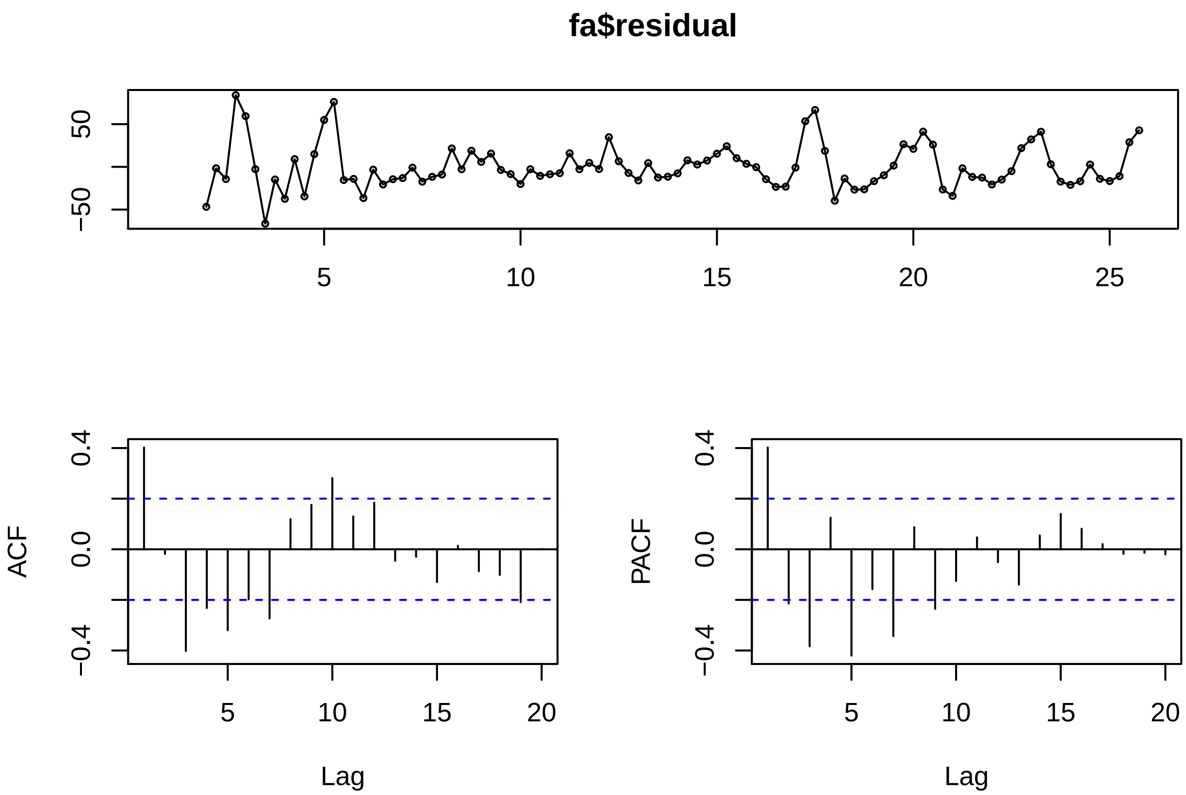

```
fo=forecast(ho)
tsdisplay(fo$residual)
```




```
Box.test(fo$residual, type="Box-Pierce")
```

```
##
##  Box-Pierce test
##
## data:  fo$residual
## X-squared = 49.863, df = 1, p-value = 1.648e-12
```

```
fa=forecast(ha)
tsdisplay(fa$residual)
```




```r
Box.test(fa$residual, type="Box-Pierce")
```

```
##
##  Box-Pierce test
##
## data:  fa$residual
## X-squared = 15.538, df = 1, p-value = 8.085e-05
```

```r
accuracy(fo)
```

```
##                     ME     RMSE      MAE  MPE MAPE      MASE      ACF1
## Training set -2.136802 36.97003 30.07774 -Inf  Inf 0.6282556 0.7207004
```

```r
accuracy(fa)
```

```
##                      ME     RMSE      MAE  MPE MAPE      MASE     ACF1
## Training set 0.05809873 25.79089 19.20619 -Inf  Inf 0.4011737 0.402316
```

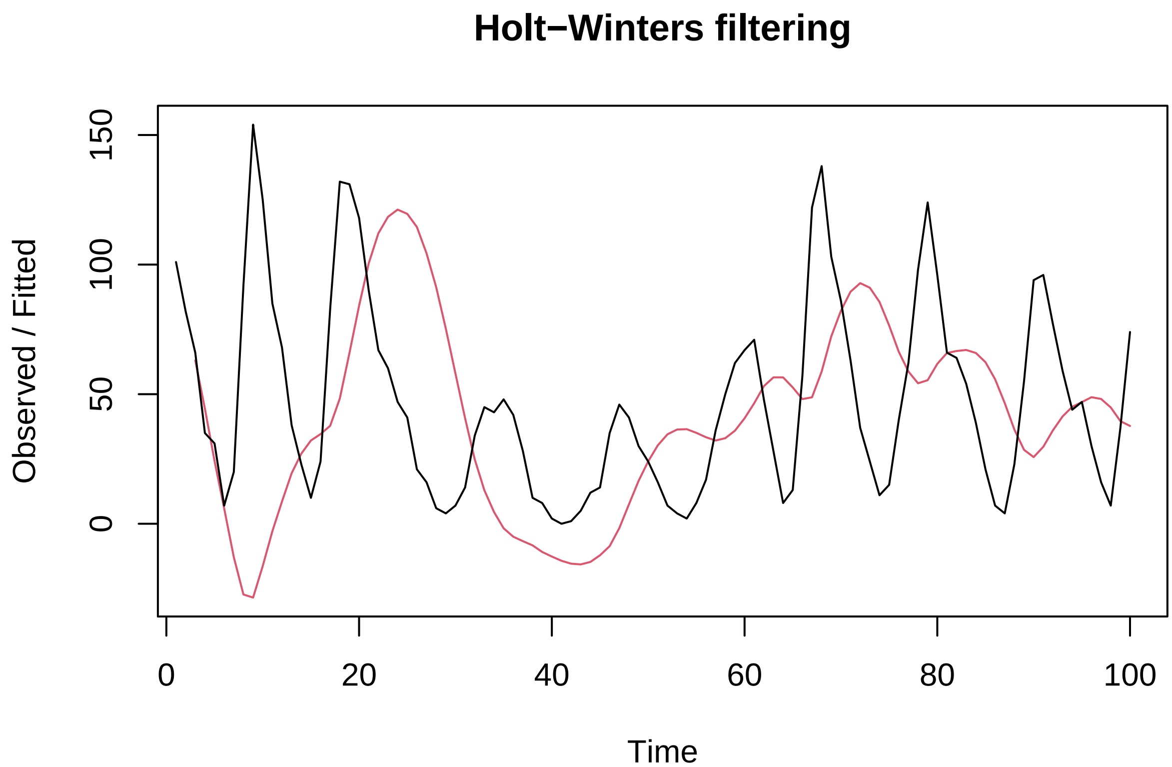
no seasonal effect

```r
ho=HoltWinters(spot.ts, alpha=0.1, gamma=F) #exponential smoothing # (gamma=F): no seasonal effect
ho
```

```
## Holt-Winters exponential smoothing with trend and without seasonal component.
##
## Call:
## HoltWinters(x = spot.ts, alpha = 0.1, gamma = F)
##
## Smoothing parameters:
##  alpha: 0.1
##  beta : 0.3835598
```

```
##   gamma: FALSE
##
## Coefficients:
##          [,1]
## a 41.3674862
## b -0.2065062
```

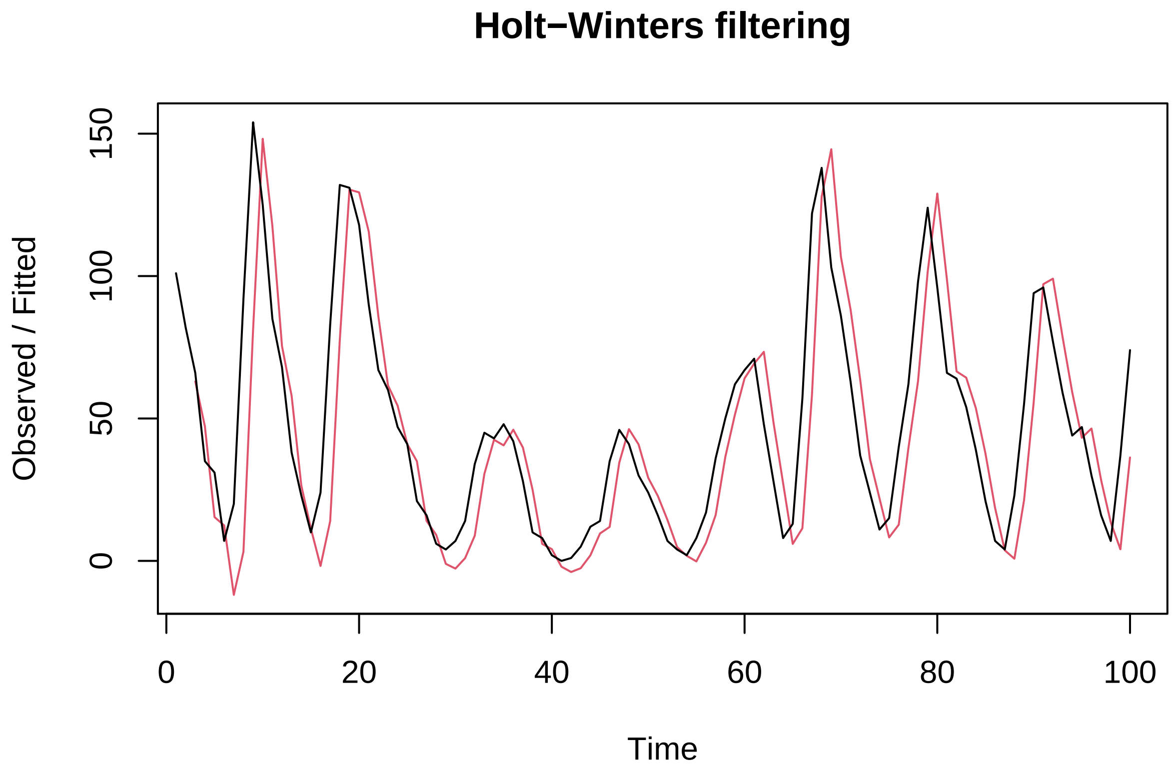
```r
plot(ho)
```



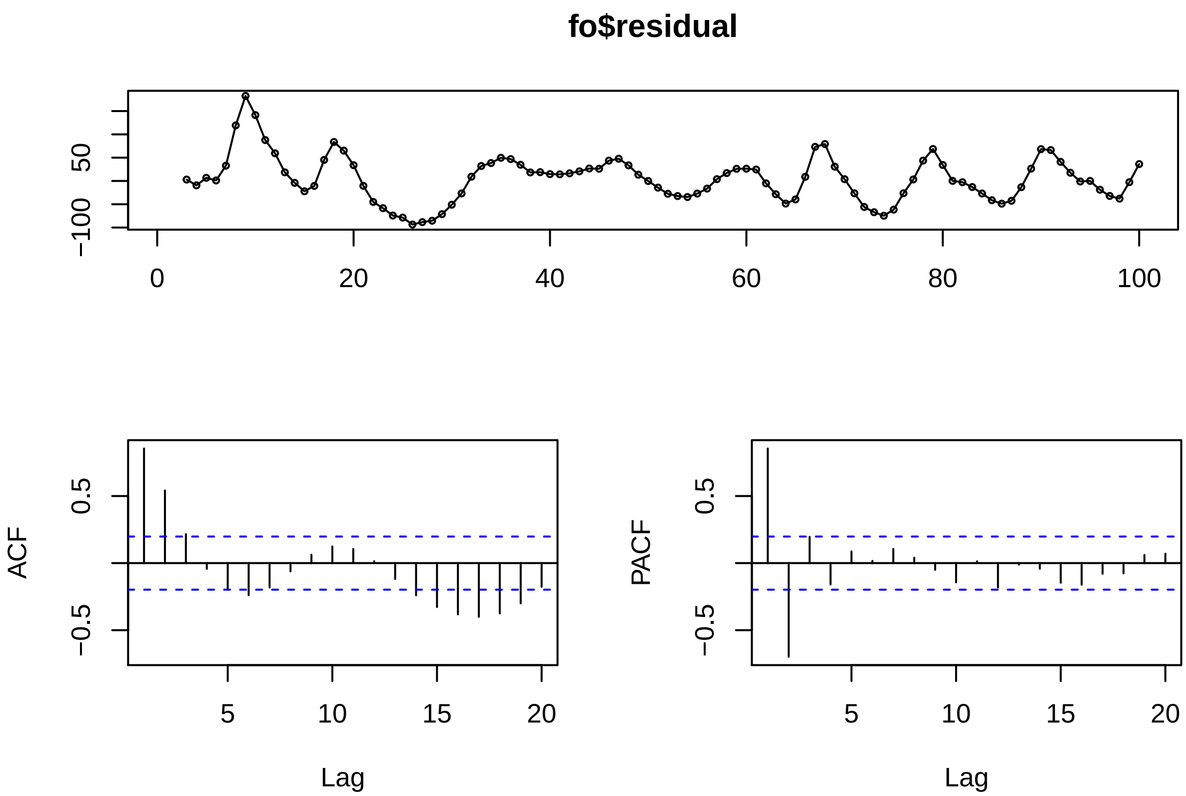

```r
ha=HoltWinters(spot.ts, gamma=F) #exponential smoothing
ha
```

```
## Holt-Winters exponential smoothing with trend and without seasonal component.
##
## Call:
## HoltWinters(x = spot.ts, gamma = F)
##
## Smoothing parameters:
##   alpha: 1
##   beta : 0.0678801
##   gamma: FALSE
##
## Coefficients:
##         [,1]
## a 74.000000
## b  1.870588
```

```r
plot(ha) # the red line is the fitted value
```



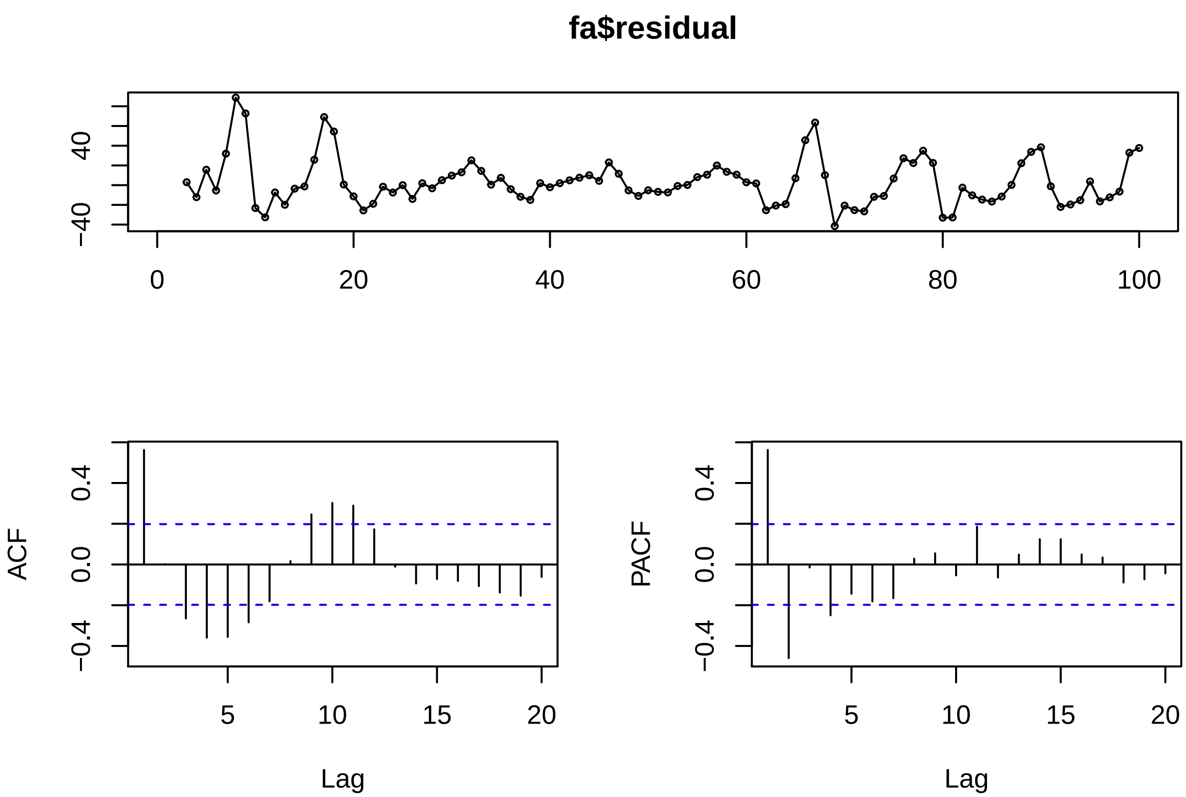

```
fo=forecast(ho)
tsdisplay(fo$residual)
```



```r
Box.test(fo$residual, type="Box-Pierce")
```

```
##
##  Box-Pierce test
##
## data:  fo$residual
## X-squared = 71.53, df = 1, p-value < 2.2e-16
```

```r
fa=forecast(ha)
tsdisplay(fa$residual)
```




```r
Box.test(fa$residual, type="Box-Pierce")
```

```
##
##  Box-Pierce test
##
## data:  fa$residual
## X-squared = 30.996, df = 1, p-value = 2.585e-08
```

```r
accuracy(fo)
```

```
##                    ME    RMSE      MAE MPE MAPE     MASE     ACF1
## Training set 4.999751 48.9669 37.57134 Inf  Inf 2.191846 0.854341
```

```r
accuracy(fa)
```

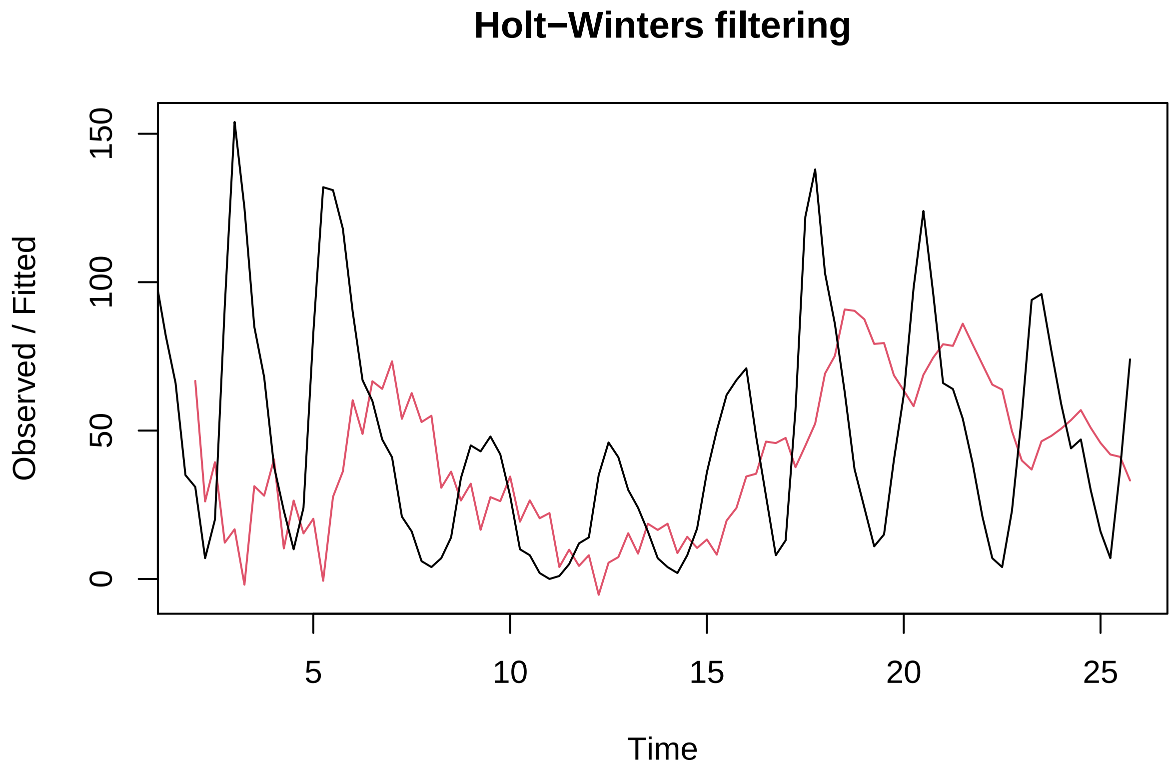
```
##                    ME     RMSE      MAE MPE MAPE      MASE      ACF1
## Training set 3.137373 23.61944 16.89736 Inf  Inf 0.9857625 0.5623951
```

trend and seasonal effect exist

```
ho=HoltWinters(spot.ts4, alpha=0.1) #exponential smoothing
ho
```

```
## Holt-Winters exponential smoothing with trend and additive seasonal component.
##
## Call:
## HoltWinters(x = spot.ts4, alpha = 0.1)
##
## Smoothing parameters:
##  alpha: 0.1
##  beta : 0.1661245
##  gamma: 0.1119542
##
## Coefficients:
##          [,1]
## a  23.503275
## b  -1.460056
## s1 10.818338
## s2 10.880312
## s3 18.749426
## s4 17.869043
```

```
plot(ho)
```

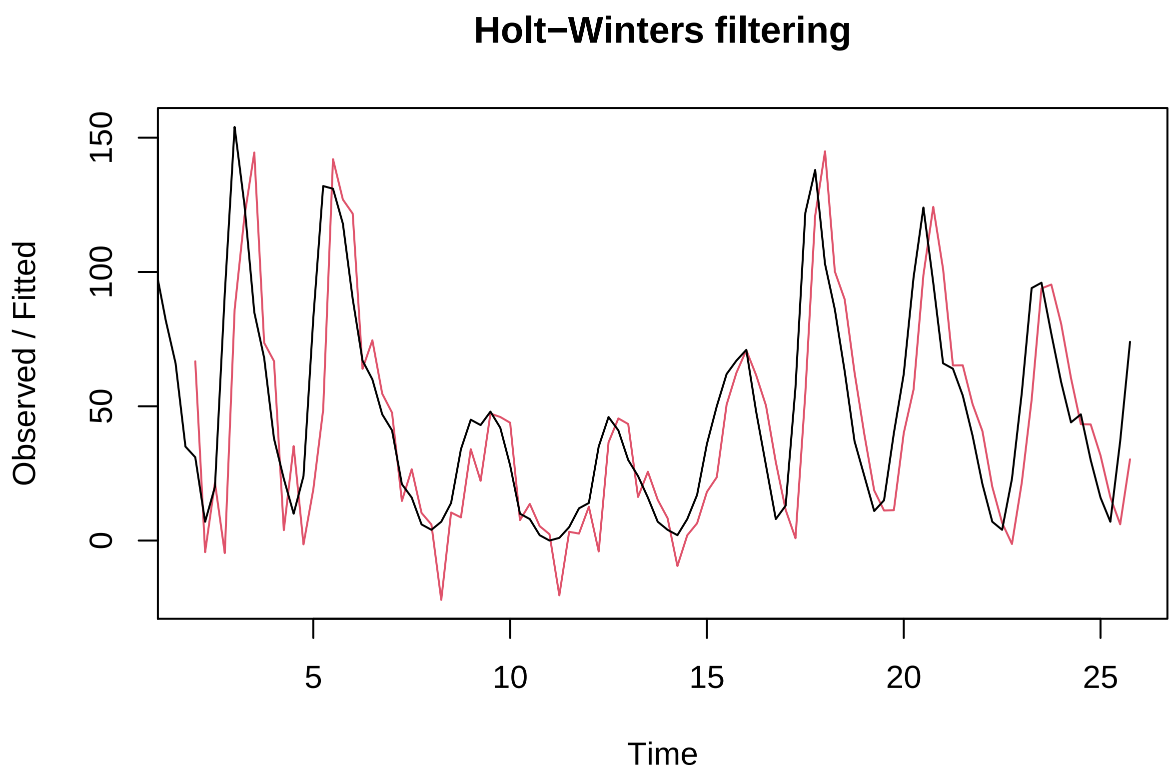



```
ha=HoltWinters(spot.ts4) #exponential smoothing
ha
```

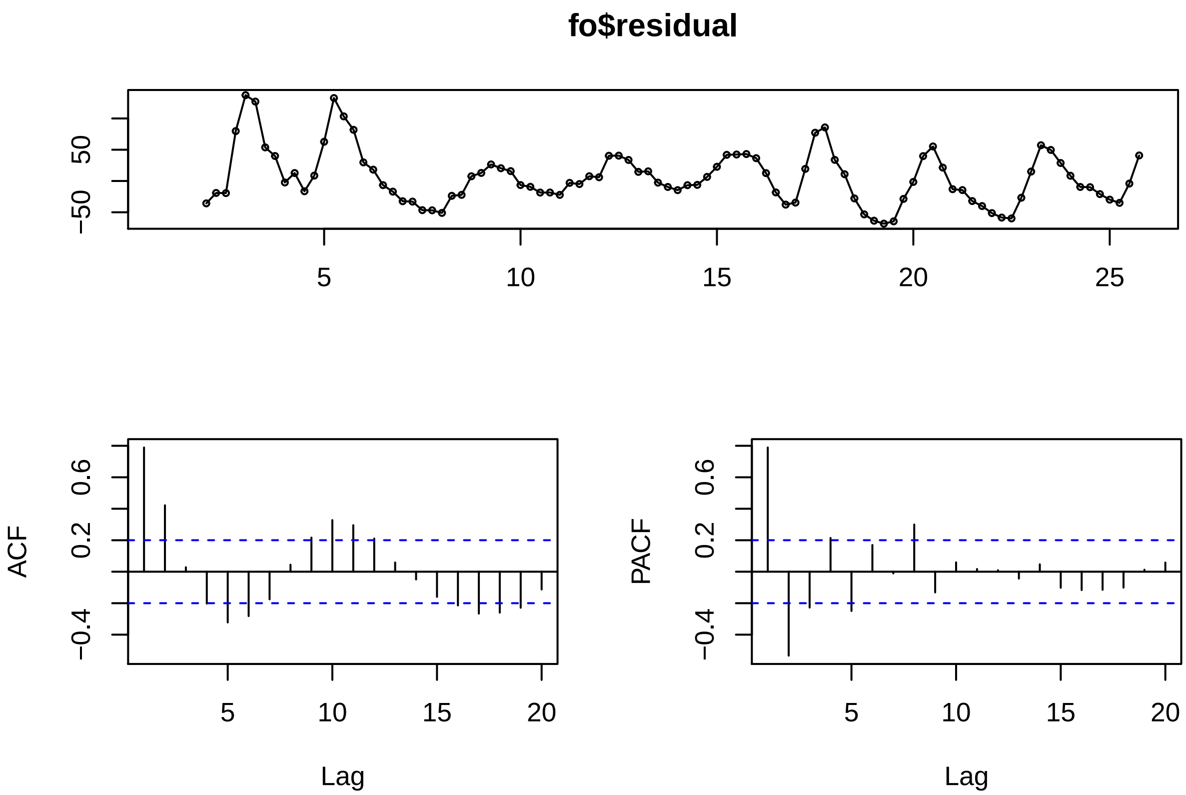
```
## Holt-Winters exponential smoothing with trend and additive seasonal component.
##
```

```
## Call:
## HoltWinters(x = spot.ts4)
##
## Smoothing parameters:
##  alpha: 0.9370146
##  beta : 0.03377394
##  gamma: 1
##
## Coefficients:
##          [,1]
## a  68.8328871
## b   0.4755106
## s1  4.2076887
## s2  5.3628840
## s3  8.2509240
## s4  5.1671129
```

```
plot(ha) # the red line is the fitted value
```



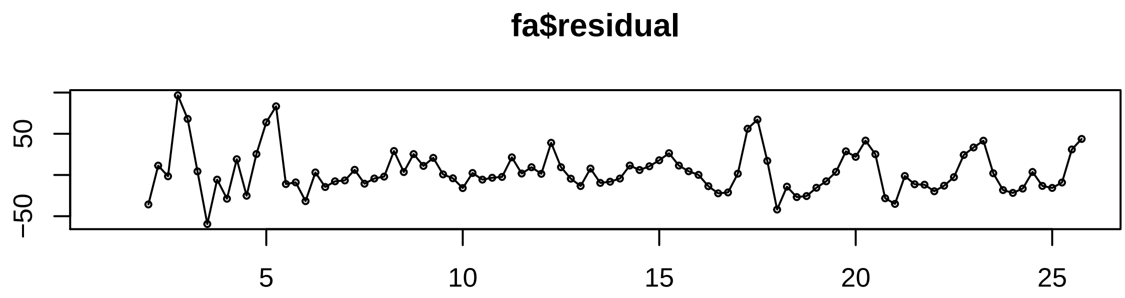

```
fo=forecast(ho)
tsdisplay(fo$residual)
```

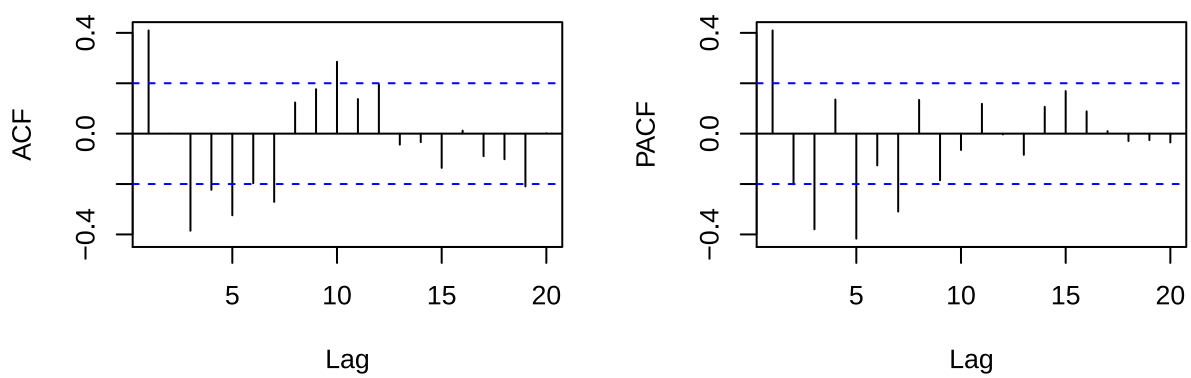



```
Box.test(fo$residual, type="Box-Pierce")
```

```
##
##  Box-Pierce test
##
## data:  fo$residual
## X-squared = 59.808, df = 1, p-value = 1.044e-14
```

```
fa=forecast(ha)
tsdisplay(fa$residual)
```






```r
Box.test(fa$residual, type="Box-Pierce")
```

```
##
##  Box-Pierce test
##
## data:  fa$residual
## X-squared = 16.099, df = 1, p-value = 6.01e-05
```

```r
accuracy(fo)
```

```
##                    ME     RMSE      MAE  MPE MAPE      MASE      ACF1
## Training set 6.028953 43.24193 33.07695 -Inf  Inf 0.6909023 0.7893013
```

```r
accuracy(fa)
```

```
##                   ME     RMSE      MAE  MPE MAPE      MASE      ACF1
## Training set 3.80191 26.57515 19.03591 -Inf  Inf 0.3976169 0.4095143
```

RMSE of the optimized alpha is always smaller than RMSE of alpha=0.1.

In case of only no trend effect and optimized alpha, residual plot doesn't seem to have pattern.

In all cases, several bars are crossing the blue line at ACF and PACF graph.

Plus, p-value is smaller than significance level (0.05) at Box-Pierce test.

Therefore, residuals isn't stationary and independence.

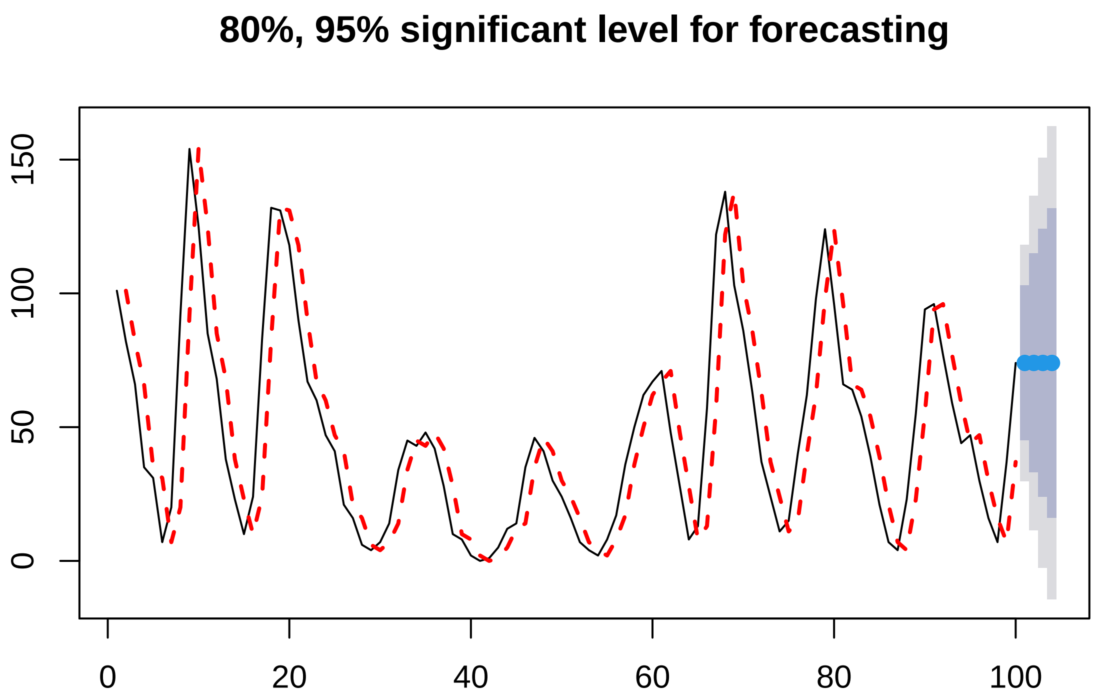
To find the best exponential smoothing model for spot data, We must compare RMSE of all models. RMSE was the lowest in the model with no trend effect, no seasonal effect and optimized alpha (0.9999339).

5.From your best model, find the forecast of next 4 points.

```
ha=HoltWinters(spot.ts,beta=F, gamma=F) #exponential smoothing
fa=forecast(ha, h=4)
fa
```

```
##     Point Forecast    Lo 80    Hi 80       Lo 95    Hi 95
## 101       73.99755 45.07885 102.9163  29.770210 118.2249
## 102       73.99755 33.10168 114.8934  11.452711 136.5424
## 103       73.99755 23.91109 124.0840  -2.603077 150.5982
## 104       73.99755 16.16301 131.8321 -14.452749 162.4479
```

```
plot(fa,main="80%, 95% significant level for forecasting")
lines(fa$fitted, col="red", lty=2, lwd=2)
```

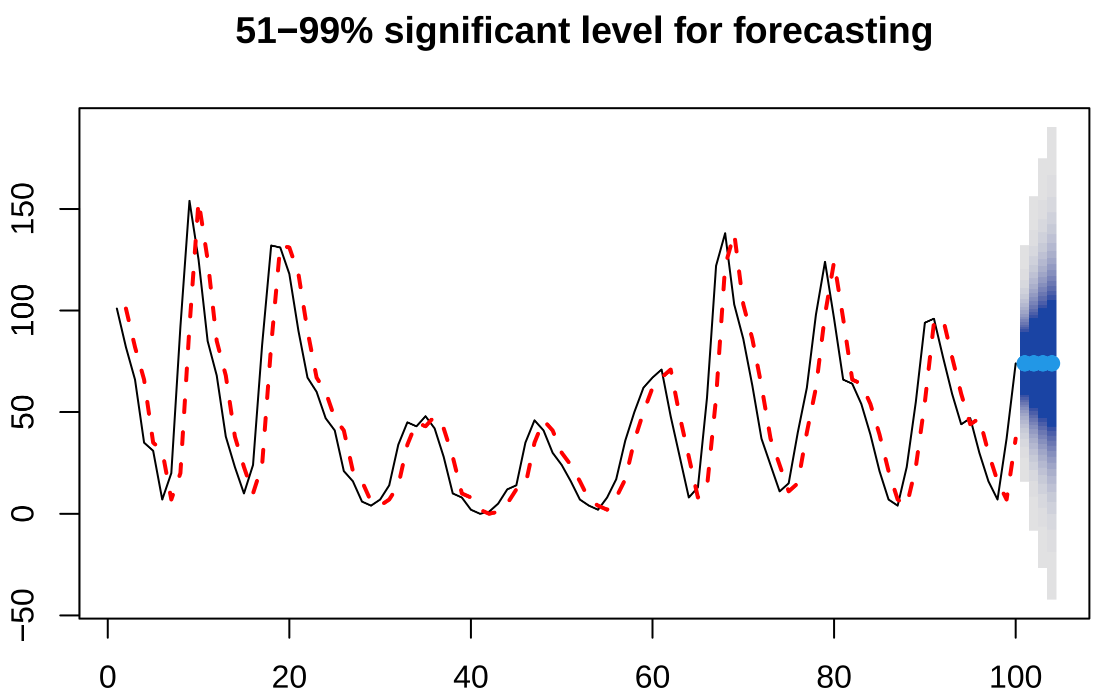



```
Fa=forecast(ha, h=4, fan=T)
Fa
```

```
##     Point Forecast    Lo 51     Hi 51    Lo 54     Hi 54    Lo 57     Hi 57
## 101       73.99755 58.42047  89.57464 57.32519  90.66992 56.18914  91.80597
## 102       73.99755 51.96896  96.02615 50.42005  97.57506 48.81349  99.18162
## 103       73.99755 47.01844 100.97667 45.12144 102.87366 43.15383 104.84127
## 104       73.99755 42.84493 105.15018 40.65448 107.34063 38.38249 109.61262
##        Lo 60     Hi 60    Lo 63     Hi 63    Lo 66     Hi 66    Lo 69     Hi 69
## 101 55.00605  92.98906 53.76829  94.22682 52.46645  95.52866 51.08868  96.90643
## 102 47.14039 100.85471 45.38999 102.60511 43.54898 104.44613 41.60058 106.39453
## 103 41.10475 106.89036 38.96098 109.03413 36.70623 111.28888 34.31996 113.67514
## 104 36.01642 111.97869 33.54102 114.45408 30.93747 117.05763 28.18207 119.81304
##        Lo 72     Hi 72    Lo 75     Hi 75    Lo 78    Hi 78    Lo 81    Hi 81
## 101 49.61973  98.37538 48.03948  99.95563 46.32047 101.6746 44.42383 103.5713
## 102 39.52325 108.47186 37.28850 110.70661 34.85754 113.1376 32.17538 115.8197
## 103 31.77579 116.21932 29.03883 118.95628 26.06156 121.9336 22.77662 125.2185
## 104 25.24432 122.75078 22.08397 125.91114 18.64614 129.3490 14.85304 133.1421
##        Lo 84    Hi 84     Lo 87    Hi 87      Lo 90    Hi 90     Lo 93    Hi 93
```

```
## 101 42.29157 105.7035 39.831260 108.1638 36.8807966 111.1143 33.111090 114.8840
## 102 29.16001 118.8351 25.680715 122.3144 21.5082674 126.4868 16.177273 131.8178
## 103 19.08360 128.9115 14.822405 133.1727  9.7122777 138.2828  3.183242 144.8119
## 104 10.58873 137.4064  5.668354 142.3268 -0.2322802 148.2274 -7.771320 155.7664
##          Lo 96    Hi 96      Lo 99    Hi 99
## 101  27.653917 120.3412  15.872971 132.1221
## 102   8.459920 139.5352  -8.200303 156.1954
## 103  -6.268443 154.2636 -26.672740 174.6678
## 104 -18.685125 166.6802 -42.245848 190.2410
```

```
plot(Fa,main="51-99% significant level for forecasting")
lines(Fa$fitted,col="red", lty=2, lwd=2)
```



The next 4 points are all 73.99755 from my best model.