Actividad-1.r

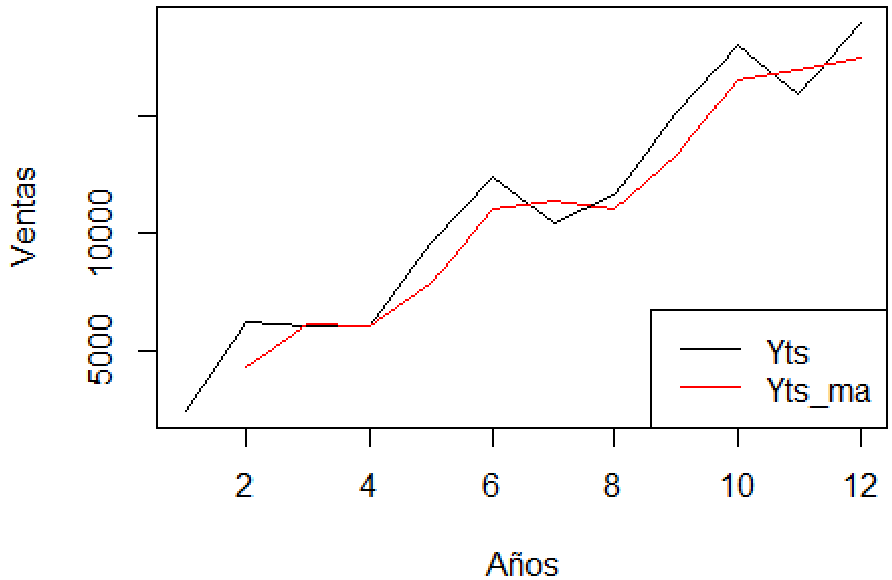
MAYE MAMANI VICTOR RAUL

2024-05-15

Las interpretaciones están de color rojo

```
#LIBRERIAS NECESARIA
library(ggplot2) #PARA HACER GRAFICOS
library(forecast) #PRONOSTICO

## Warning: package 'forecast' was built under R version 4.3.3

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

library(TTR) #metodo de media movil
library(readxl) #para leer archivos excel
library(fpp2)

## Warning: package 'fpp2' was built under R version 4.3.3

## ── Attaching packages ────────────────────────────────────────────── f
pp2 2.5 ──

## ✓ fma       2.5      ✓ expsmooth 2.3

## Warning: package 'fma' was built under R version 4.3.3

## Warning: package 'expsmooth' was built under R version 4.3.3

##

#EJERCICIO 1


x <- c(1,2,3,4,5,6,7,8,9,10,11,12)
y <- c(2400,6200,6000,6000,9600,12400,10400,11600,15200,18000,16000,19000
)
datos <- data.frame(Años = x, Ventas = y)

View(datos)

Yts <- ts(datos$Ventas, start = c(1,1), frequency = 1)
print(Yts)

## Time Series:
## Start = 1
```

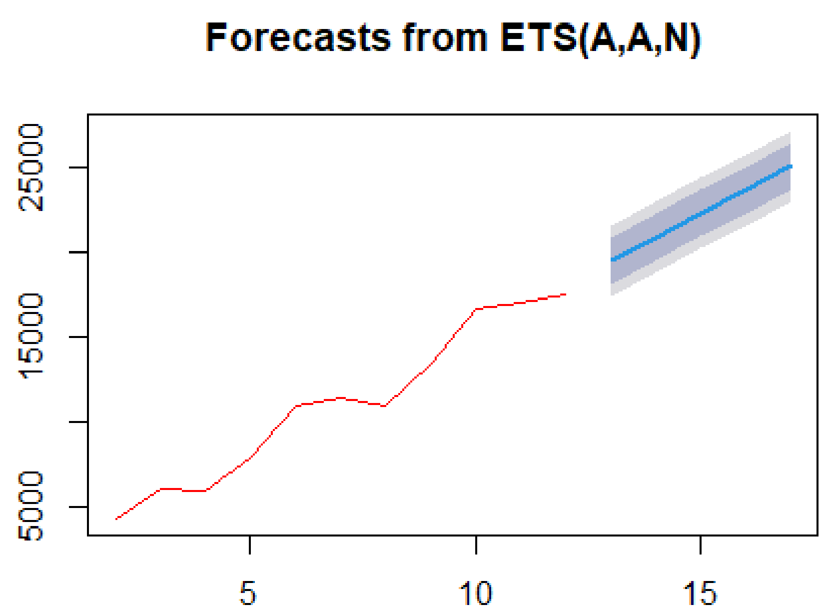
```
## End = 12
## Frequency = 1
##  [1]  2400  6200  6000  6000  9600 12400 10400 11600 15200 18000 16000
19000

#Grafico de la serie ORIGINAL
plot(Yts, type="l", xlab = "Años", ylab="Ventas")

################################
#MEDIA MOVIL SIMPLE
################################

library(TTR) #metodo de media movil
k<-2 #Periodo
Yts_ma <- SMA(Yts, k)
print(Yts_ma)

## Time Series:
## Start = 1
## End = 12
## Frequency = 1
##  [1]    NA  4300  6100  6000  7800 11000 11400 11000 13400 16600 17000
17500

#Grafico de la serie original y la media movil simple
plot(Yts, type="l", xlab = "Años", ylab="Ventas")
lines(Yts_ma, type="l", col="red")
legend(x = "bottomright", legend =  c("Yts", "Yts_ma"), col = c("black","
red"), lty=c(1,1))
```




```
# Grafico de la predicción de la media movil simple
pred <- predict(Yts_ma, h=5)

## Warning in ets(object, lambda = lambda, biasadj = biasadj,
## allow.multiplicative.trend = allow.multiplicative.trend, : Missing val
ues
## encountered. Using longest contiguous portion of time series

summary(pred)

##
## Forecast method: ETS(A,A,N)
##
## Model Information:
## ETS(A,A,N)
##
## Call:
##  ets(y = object, lambda = lambda, biasadj = biasadj, allow.multiplicat
ive.trend = allow.multiplicative.trend)
##
##   Smoothing parameters:
##     alpha = 0.0018
##     beta  = 0.0014
##
##   Initial states:
##     l = 2711.9151
##     b = 1396.1517
```

```
##
##   sigma:  1062.164
##
##      AIC     AICc      BIC
## 184.7024 196.7024 186.6919
##
## Error measures:
##                    ME     RMSE     MAE        MPE    MAPE      MASE
ACF1
## Training set 9.758719 847.3138 727.044 -0.2851712 7.32242 0.5120028 -0
.04582012
##
## Forecasts:
##    Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## 13       19468.14 18106.92 20829.36 17386.34 21549.94
## 14       20864.44 19503.21 22225.66 18782.62 22946.25
## 15       22260.73 20899.50 23621.97 20178.90 24342.57
## 16       23657.03 22295.77 25018.29 21575.16 25738.90
## 17       25053.33 23692.03 26414.62 22971.40 27135.25
```

```r
plot(pred, col="red")
```

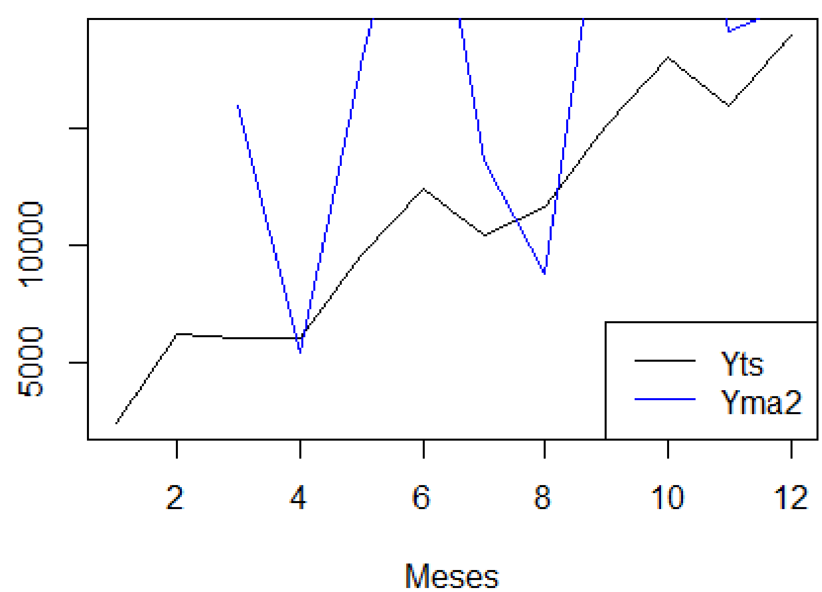



```r
###########################
#MEDIA MOVIL DOBLE
###########################

Yts_ma2 <- SMA(Yts_ma, k) # Media móvil de media móvil
```

```
a <- 2*Yts_ma - Yts_ma2
b <- (2/(k-1))*(Yts_ma - Yts_ma2)
p <- 5 # Periodo futuro a pronosticar
Yma2 <- a + b*p
print(Yma2)
```

```
## Time Series:
## Start = 1
## End = 12
## Frequency = 1
##  [1]    NA    NA 16000  5450 17700 28600 13600  8800 26600 34200 19200
20250
```

```
# Graficar la serie original y la media móvil doble
plot(Yts, type = "l", xlab="Meses", ylab=" ") # Serie original
lines(Yma2, type = "l", col = "blue") # Media móvil doble
legend(x = "bottomright", legend = c("Yts", "Yma2"), col = c('black', 'bl
ue'), lty = c(1,
1))
```



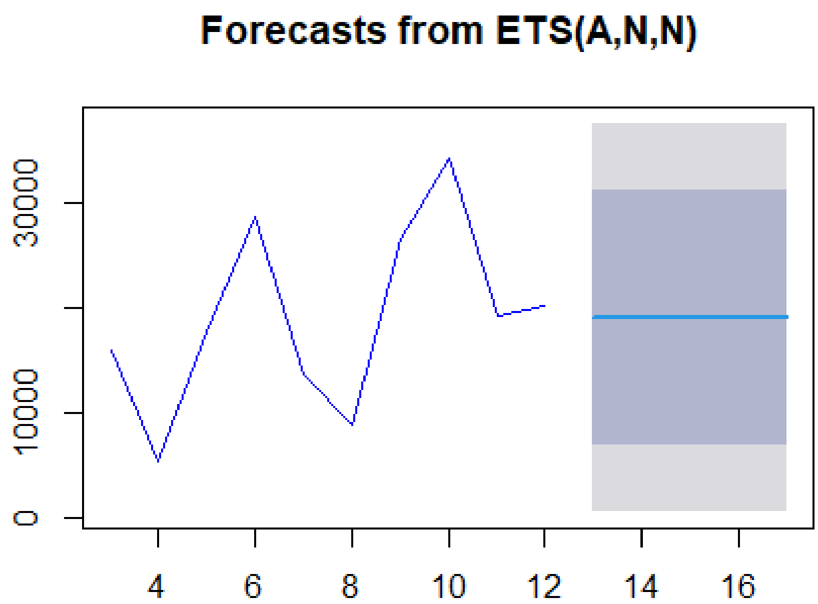

```
# Graficar la predicción de la media movil doble
pred <- predict(Yma2, h=5)
```

```
## Warning in ets(object, lambda = lambda, biasadj = biasadj,
## allow.multiplicative.trend = allow.multiplicative.trend, : Missing val
ues
## encountered. Using longest contiguous portion of time series
```

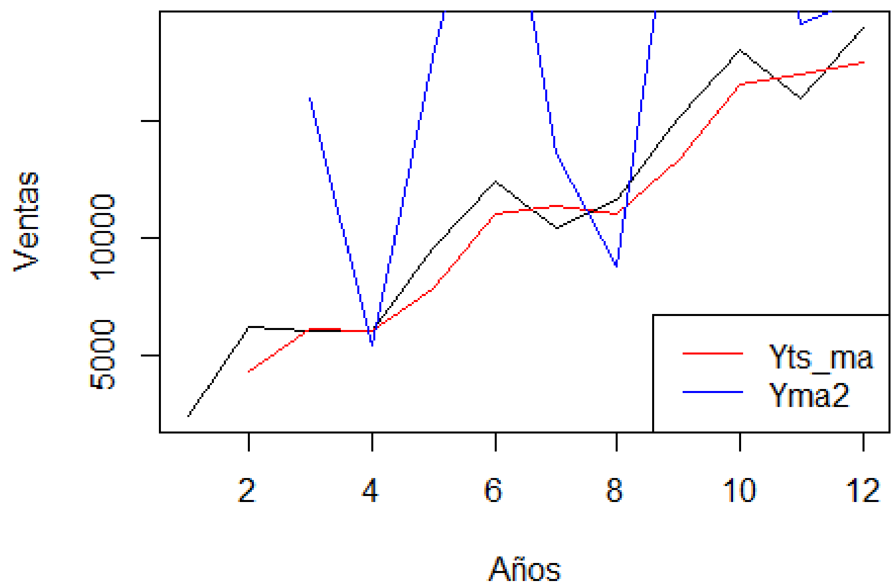
```r
summary(pred)
```

```
##
## Forecast method: ETS(A,N,N)
##
## Model Information:
## ETS(A,N,N)
##
## Call:
##  ets(y = object, lambda = lambda, biasadj = biasadj, allow.multiplicat
ive.trend = allow.multiplicative.trend)
##
##   Smoothing parameters:
##     alpha = 1e-04
##
##   Initial states:
##     l = 19039.0358
##
##   sigma:  9421.452
##
##      AIC     AICc      BIC
## 209.8093 213.8093 210.7171
##
## Error measures:
##                    ME     RMSE      MAE       MPE     MAPE      MASE
ACF1
## Training set 1.994431 8426.803 6730.192 -31.91832 54.52328 0.6379329 0
.1270712
##
## Forecasts:
##    Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## 13       19039.04 6964.962 31113.11 573.3319 37504.74
## 14       19039.04 6964.962 31113.11 573.3318 37504.74
## 15       19039.04 6964.962 31113.11 573.3317 37504.74
## 16       19039.04 6964.961 31113.11 573.3316 37504.74
## 17       19039.04 6964.961 31113.11 573.3315 37504.74
```

```r
plot(pred, col="blue")
```



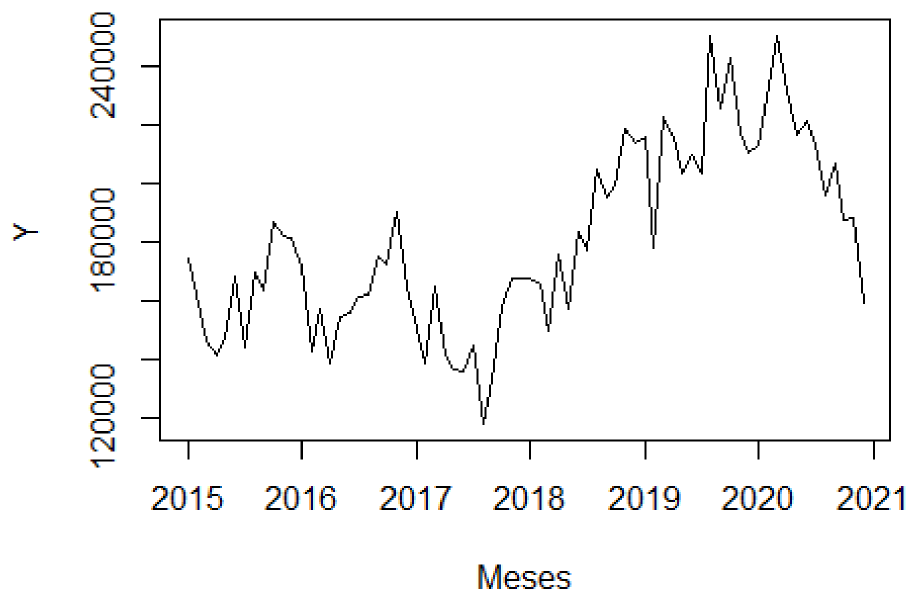

```
#GRAFICA DE AMBOS METODOS con la serie original
plot(Yts, type='l', xlab="Años", ylab="Ventas")
lines(Yts_ma, type='l', col="red")
lines(Yma2, type='l', col="blue")
legend(x = "bottomright", legend = c("Yts_ma", "Yma2"), col = c('red', 'b
lue'), lty = c(1, 1))
```



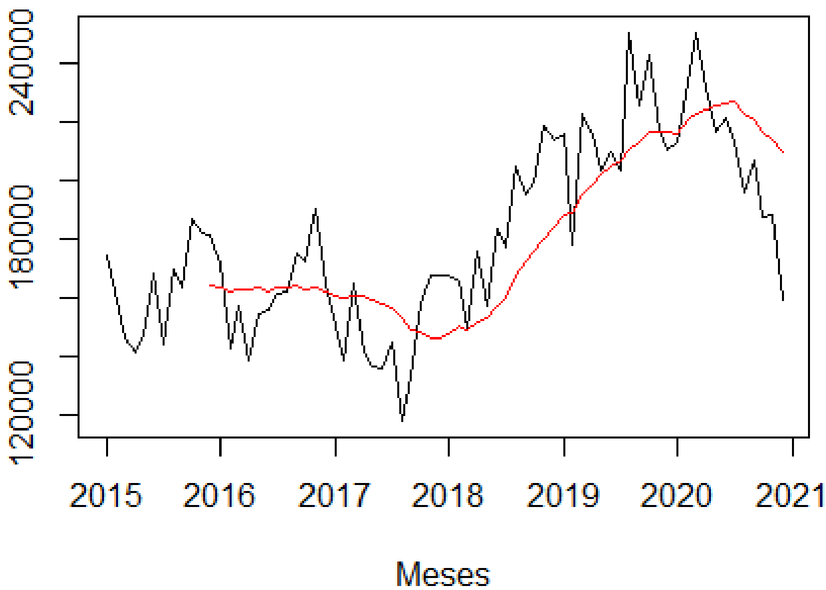

```
# La media movil simple se ajusta mas a los datos originales que l
a media movil doble.

#################################################
#EJERCICIO 2


datos = read_excel("F:/777--Programacion repos/Una/r/data/02-t.xlsx")

Yts <- ts(datos$Ventas, start = c(2015,1), frequency = 12)
print(Yts)

##          Jan    Feb    Mar    Apr    May    Jun    Jul    Aug    Sep
Oct
## 2015 174132 161943 146195 141146 147081 168127 144123 169350 163300 18
6654
## 2016 171695 142800 156978 138780 154502 155887 161107 162323 175434 17
2045
## 2017 150419 138585 164806 141578 136450 136144 144638 118136 133284 15
8266
## 2018 167654 165798 149714 175504 156909 183660 177009 204359 194735 19
9345
## 2019 215773 177886 222408 215793 203316 209376 203557 250105 225476 24
2739
## 2020 212658 232616 249744 231813 216286 220976 212788 195399 207013 18
7727
```

```
##          Nov    Dec
## 2015 182168 181453
## 2016 190468 164692
## 2017 167523 167508
## 2018 218171 213791
## 2019 217358 210469
## 2020 188220 159060
```

```r
#Graficar la serie temporal
plot(Yts,type = "l", xlab="Meses",ylab="Y")
```



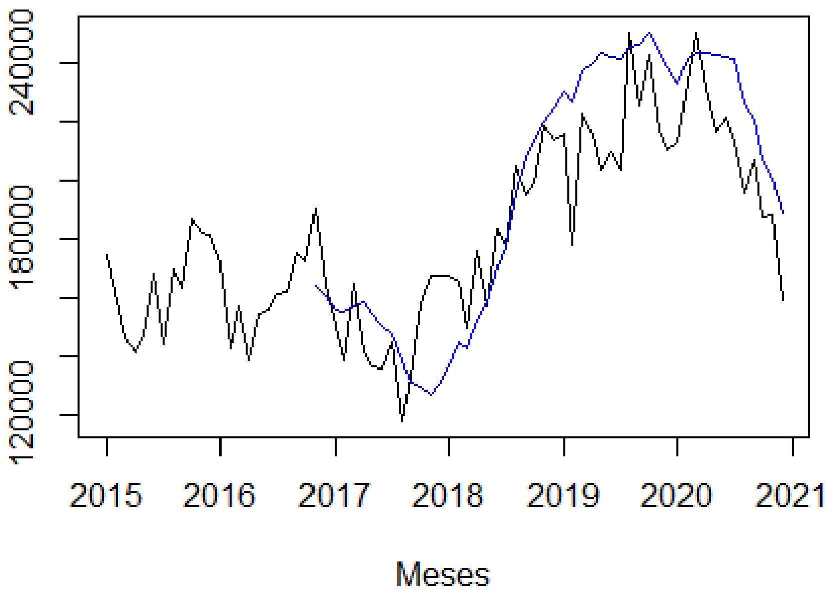

```r
# a) Construir un promedio móvil de orden 12
k = 12
Yts_ma <- SMA(Yts,k)

# Graficar la serie temporal con la media movil
plot(Yts, type = "l", xlab = "Meses", ylab = " ") # Serie original
lines(Yts_ma, type = "l", col = "red")
```



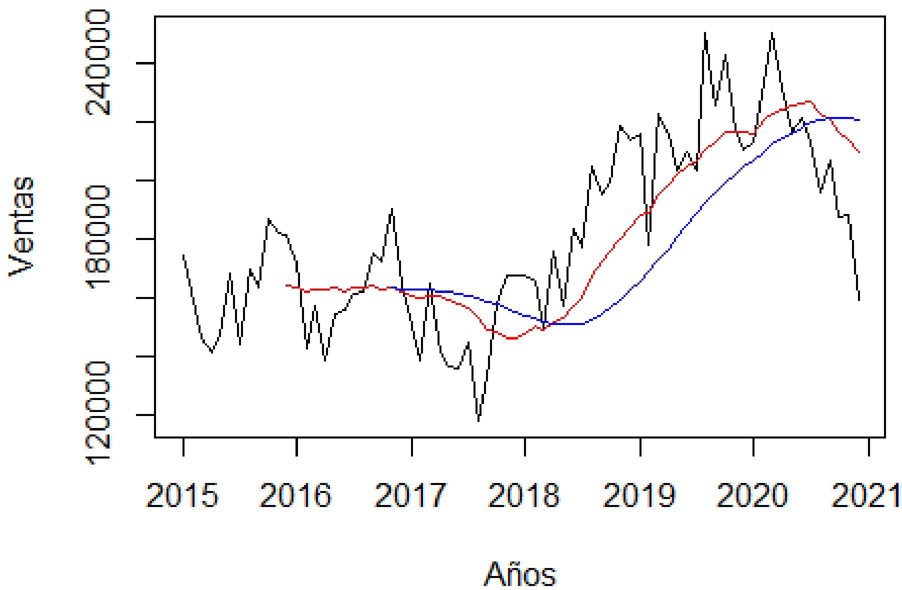

```
# b) Construir un promedio móvil doble de orden 12
Yts_ma2 <- SMA(Yts_ma, k) # Media móvil de media móvil
a <- 2*Yts_ma - Yts_ma2
b <- (2/(k-1))*(Yts_ma - Yts_ma2)
p <- 5 # Periodo futuro a pronosticar
Yma2 <- a + b*p
print(Yma2)
```

```
##           Jan      Feb      Mar      Apr      May      Jun      Jul
Aug
## 2015       NA       NA       NA       NA       NA       NA       NA
NA
## 2016       NA       NA       NA       NA       NA       NA       NA
NA
## 2017 155988.2 155269.6 157509.7 158461.9 154697.4 150622.3 147784.4 13
8719.7
## 2018 136463.4 144643.9 142872.5 152571.6 158495.4 170086.8 177359.1 19
5957.6
## 2019 230191.0 226955.8 237249.0 239602.1 243094.4 241859.7 240903.9 24
5138.7
## 2020 233032.1 241275.2 243478.6 243260.5 242746.3 242087.2 241083.3 22
5911.2
##           Sep      Oct      Nov      Dec
## 2015       NA       NA       NA       NA
## 2016       NA       NA 164456.8 160644.9
## 2017 130868.5 129884.1 127091.8 130285.1
## 2018 207177.3 212730.9 219628.7 224892.1
```

```
## 2019 246143.2 250184.3 244190.9 238246.8
## 2020 220177.7 206888.5 200247.2 188844.4

# Graficar la serie original y la media móvil doble
plot(Yts, type = "l", xlab="Meses", ylab=" ") # Serie original
lines(Yma2, type = "l", col = "blue") # Media móvil doble
```



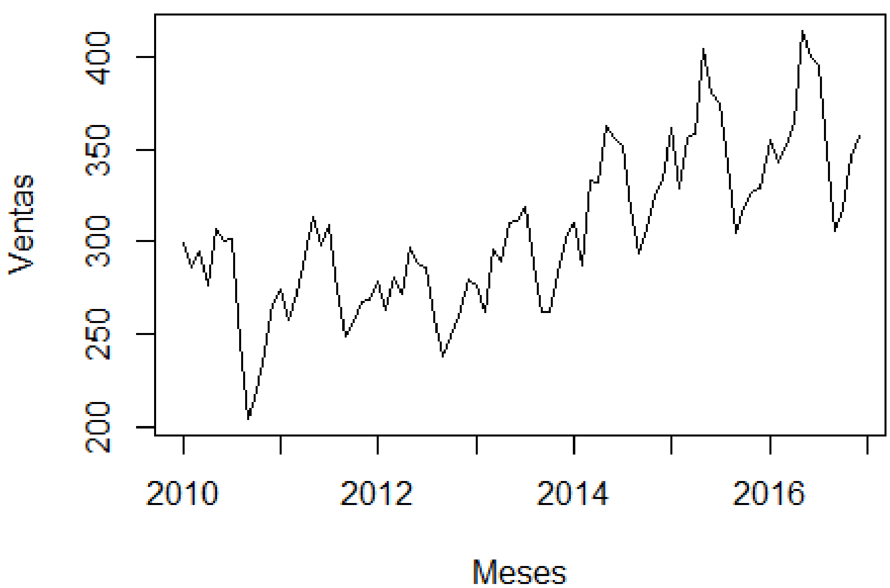

```
# c) Representar gráficamente los resultados obtenidos junto con los dato
s originales
# y comparar los resultados.

plot(Yts, type='l', xlab="Años", ylab="Ventas")
lines(Yts_ma, type='l', col="red")
lines(Yts_ma2, type='l', col="blue")
```



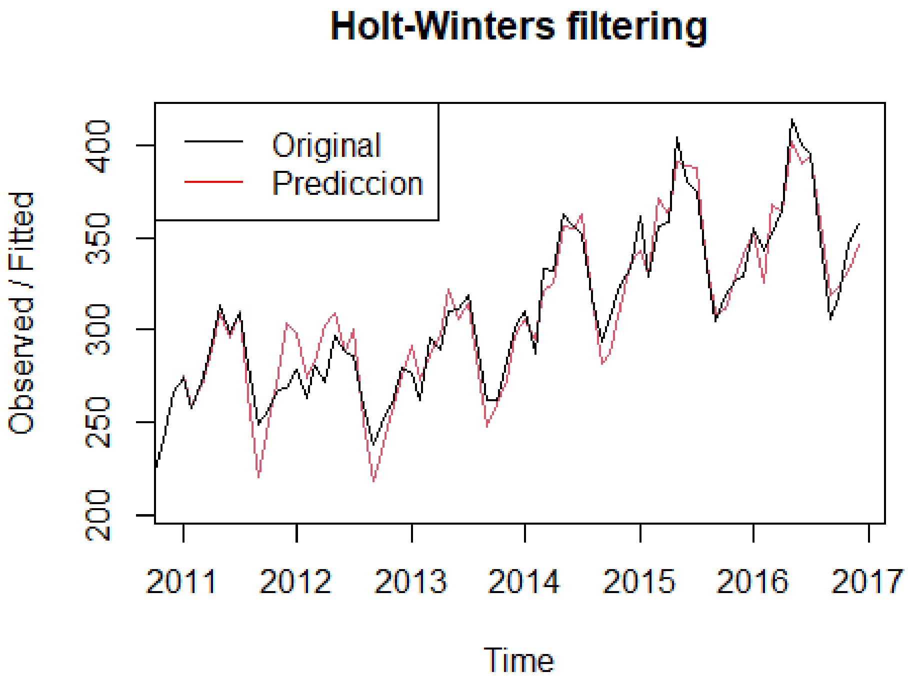

```
# Podemos ver que la media movil doble es mas suave que la media m
ovil simple.
# Y que la media movil simple se ajusta mas a los datos originales
.


##################################################
#EJERCICIO 3


datos = read_excel("F:/777--Programacion repos/Una/r/data/03-t.xlsx")

VENts <- ts(datos$Ventas, start = c(2010,1), frequency = 12)
print(VENts)
```

```
##         Jan    Feb    Mar    Apr    May    Jun    Jul    Aug    Sep
Oct
## 2010 299.53 286.04 295.52 276.49 307.33 300.94 301.97 245.94 204.29 21
9.19
## 2011 274.01 258.24 271.92 293.04 313.58 298.54 308.95 273.84 249.39 25
6.90
## 2012 278.92 263.25 281.32 272.24 296.85 288.45 286.65 258.45 237.95 24
9.31
## 2013 276.85 262.53 296.28 290.02 309.90 310.98 319.38 288.26 262.63 26
2.60
## 2014 310.87 287.53 333.20 332.65 363.05 356.85 351.80 318.12 293.49 30
```

```
7.12
## 2015 362.34 329.38 356.09 358.20 404.46 380.43 374.86 341.56 304.81 31
8.78
## 2016 355.34 343.17 352.04 364.69 414.46 399.64 395.91 348.26 306.35 31
8.43
##          Nov    Dec
## 2010 240.22 266.30
## 2011 267.29 268.80
## 2012 259.60 279.73
## 2013 283.52 301.79
## 2014 324.63 333.66
## 2015 327.14 328.93
## 2016 346.89 356.96

#Graficar la serie temporal
plot(VENts,type = "l", xlab="Meses",ylab="Ventas",col="black")
```

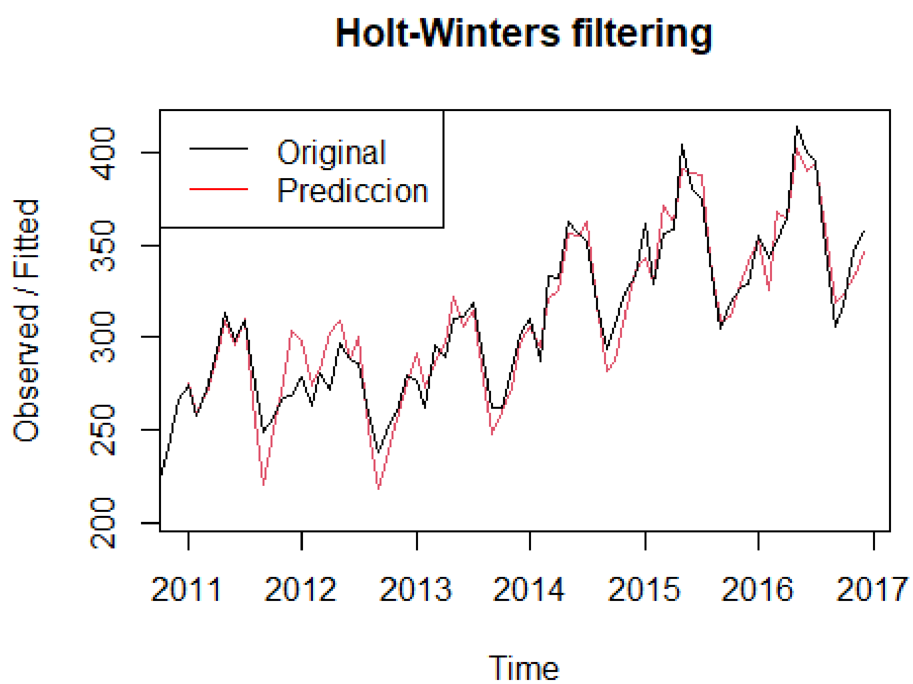



```
# HOLT WINTERS SIN PARAMETROS

VENhw_m <- HoltWinters(VENts,seasonal = "multiplicative")
plot(VENhw_m)
legend("topleft",legend = c("Original","Prediccion"),col = c("black","red
"),lty = c(1,1))
```

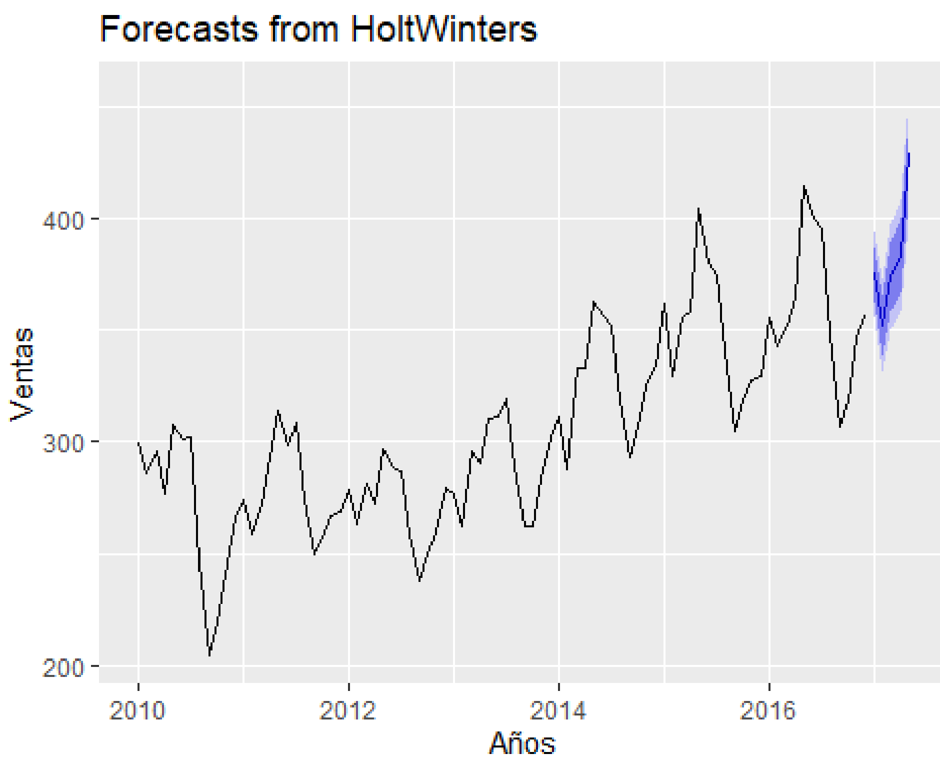



```
VENhw_m #Muestra los valores de alpha, beta y gamma

## Holt-Winters exponential smoothing with trend and multiplicative seaso
nal component.
##
## Call:
## HoltWinters(x = VENts, seasonal = "multiplicative")
##
## Smoothing parameters:
##  alpha: 0.3944988
##  beta : 0
##  gamma: 0.7054837
##
## Coefficients:
##             [,1]
## a   365.8740918
## b     0.9670600
## s1    1.0226906
## s2    0.9563513
## s3    1.0134063
## s4    1.0370153
## s5    1.1565271
## s6    1.1032888
## s7    1.0916615
## s8    0.9749303
## s9    0.8700332
## s10   0.9032878
```

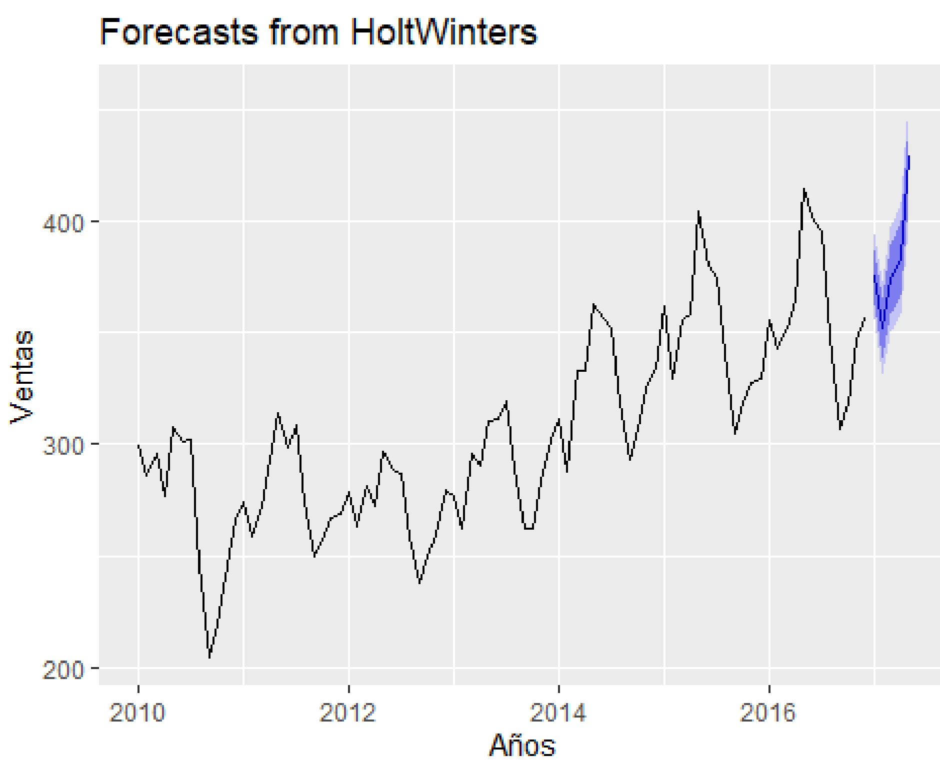
```
## s11   0.9544848
## s12   0.9706199

# HOLT WINTERS CON PARAMETROS
VENhw_m2 <- HoltWinters(VENts,seasonal = "multiplicative",optim.start = c
(0.4,0.6002,1))
plot(VENhw_m2)
legend("topleft",legend = c("Original","Prediccion"),col = c("black","red
"),lty = c(1,1))
```




```
# Predicciones

autoplot(forecast(VENhw_m,h=5),xlab="Años",ylab="Ventas",col="red")
```




```
autoplot(forecast(VENhw_m2,h=5),xlab="Años",ylab="Ventas",col="red")
```




```
print(forecast(VENhw_m,h=5))
```

```
##          Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## Jan 2017       375.1650 362.7145 387.6155 356.1236 394.2063
## Feb 2017       351.7539 338.1061 365.4016 330.8814 372.6263
## Mar 2017       373.7192 358.5460 388.8924 350.5138 396.9245
## Apr 2017       383.4285 366.9607 399.8962 358.2432 408.6137
## May 2017       428.7354 410.1895 447.2813 400.3719 457.0990
```

```r
print(forecast(VENhw_m2,h=5))
```

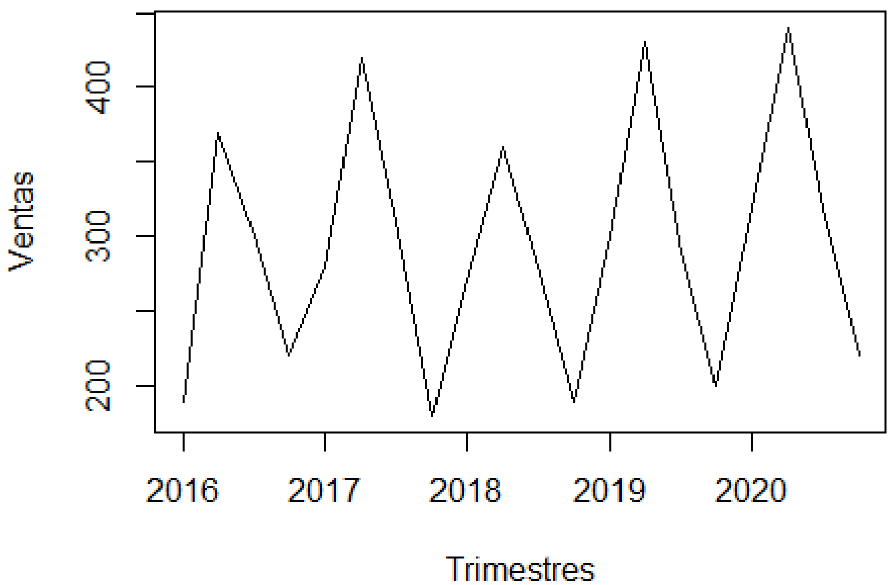
```
##          Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## Jan 2017       375.1650 362.7145 387.6155 356.1236 394.2064
## Feb 2017       351.7539 338.1061 365.4017 330.8813 372.6264
## Mar 2017       373.7191 358.5459 388.8923 350.5137 396.9245
## Apr 2017       383.4284 366.9606 399.8962 358.2431 408.6138
## May 2017       428.7355 410.1895 447.2814 400.3718 457.0991
```

 # Podemos ver que ambos metodos de Holt Winters se ajusta n a los datos originales.

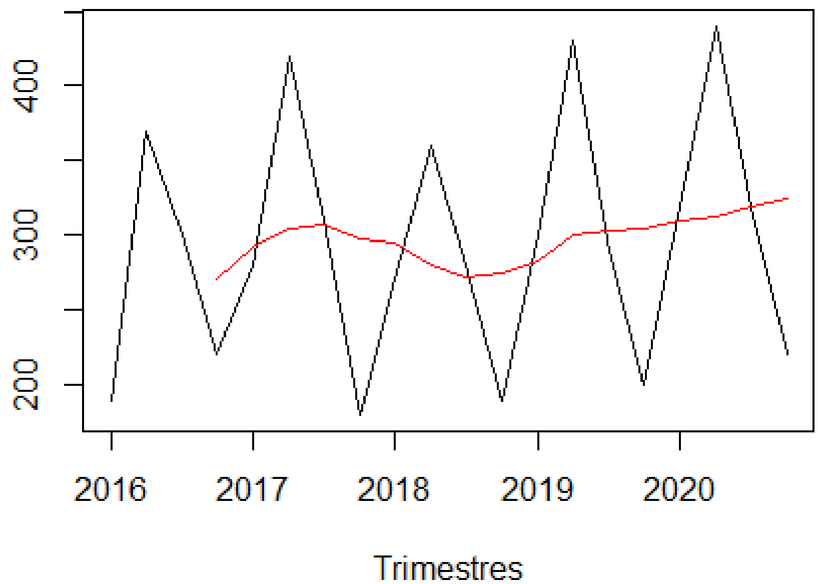
```r
##################################################
#EJERCICIO 4

datos = read_excel("F:/777--Programacion repos/Una/r/data/04-t.xlsx")

Yts <- ts(datos$Ventas, start = c(2016,1), frequency = 4)
print(Yts)
```

```
##      Qtr1 Qtr2 Qtr3 Qtr4
## 2016  190  370  300  220
## 2017  280  420  310  180
## 2018  270  360  280  190
## 2019  300  430  290  200
## 2020  320  440  320  220
```

```r
#Graficar la serie temporal
plot(Yts,type = "l", xlab="Trimestres",ylab="Ventas")
```



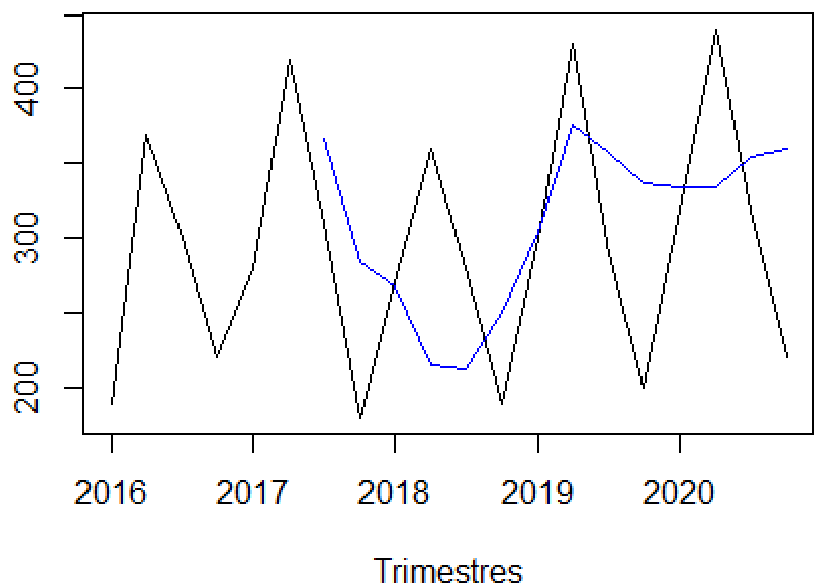

```r
# (a) Realice el pronóstico utilizando métodos de promedios móviles y det
ermine que método es
# mejor, evaluar gráficamente y utilizando los estadísticos del error.

# Promedio móvil simple
k = 4
Yts_ma <- SMA(Yts,k)

# Graficar la serie temporal original con la media movil
plot(Yts, type = "l", xlab = "Trimestres", ylab = " ") # Serie original
lines(Yts_ma, type = "l", col = "red")
```




```
# Estadísticos del error
print('Error promedio móvil simple')

## [1] "Error promedio móvil simple"

error <- Yts - Yts_ma
error_cuadrado <- error^2
mse <- mean(error_cuadrado, na.rm = TRUE)
print(paste("MSE:", mse))

## [1] "MSE: 5868.75"

rmse <- sqrt(mse)
print(paste("RMSE:", rmse))

## [1] "RMSE: 76.6077672302228"

# Promedio móvil doble

Yts_ma2 <- SMA(Yts_ma, k) # Media móvil de media móvil
a <- 2*Yts_ma - Yts_ma2
b <- (2/(k-1))*(Yts_ma - Yts_ma2)
p <- 5 # Periodo futuro a pronosticar
Yma2 <- a + b*p
print(Yma2)

##          Qtr1     Qtr2     Qtr3     Qtr4
## 2016       NA       NA       NA       NA
```

```
## 2017        NA       NA 367.0833 283.9583
## 2018 267.9167 215.0000 212.9167 250.6250
## 2019 304.1667 375.8333 356.6667 337.5000
## 2020 334.3750 334.1667 355.2083 360.2083
```

```r
# Graficar la serie original y la media móvil doble
plot(Yts, type = "l", xlab="Trimestres", ylab=" ") # Serie original
lines(Yma2, type = "l", col = "blue") # Media móvil doble
```




```r
# Estadísticos del error

print('Error promedio móvil doble')
```

```
## [1] "Error promedio móvil doble"
```

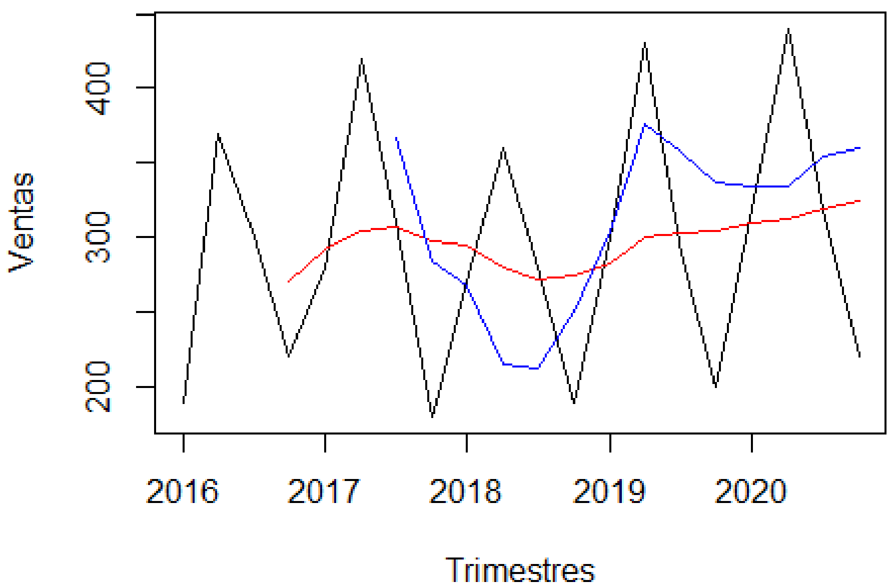
```r
error_2 <- Yts - Yts_ma2
error_2_cuadrado <- error_2^2
mse_2 <- mean(error_2_cuadrado, na.rm = TRUE)
print(paste("MSE:", mse_2))
```

```
## [1] "MSE: 6234.84933035714"
```

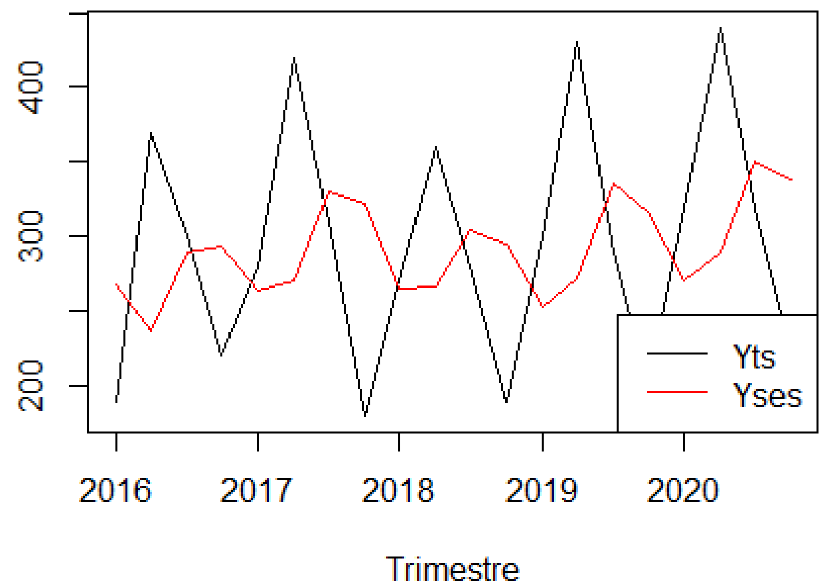
```r
rmse_2 <- sqrt(mse_2)
print(paste("RMSE:", rmse_2))
```

```
## [1] "RMSE: 78.9610621151789"
```

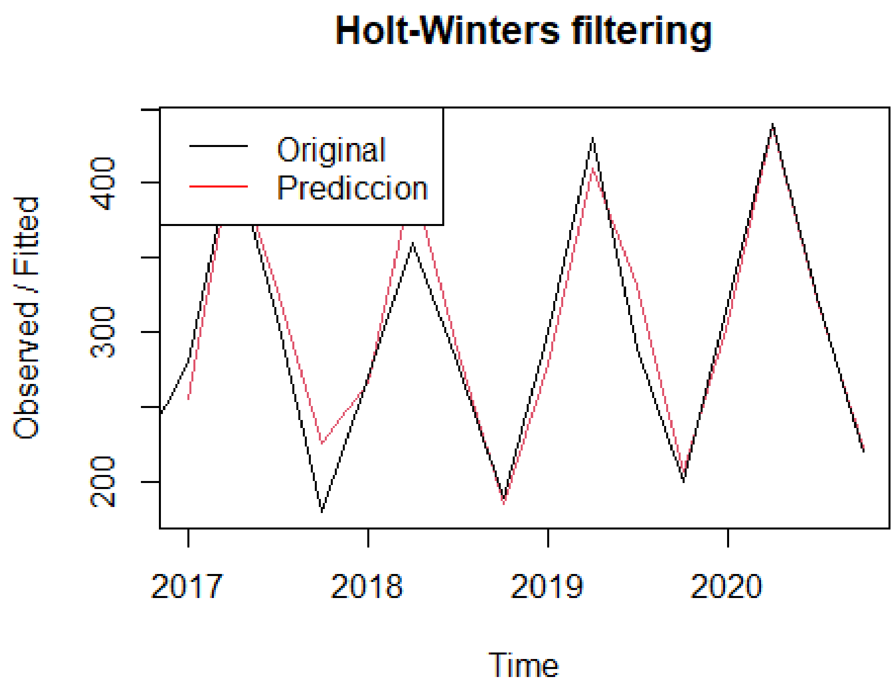
```r
# Grafica comparativa entre los 2 metodos con la serie original
plot(Yts, type = "l", xlab="Trimestres", ylab="Ventas") # Serie original
lines(Yts_ma, type = "l", col = "red") # Media móvil simple
lines(Yma2, type = "l", col = "blue") # Media móvil doble
```




```r
# Concluimos que el método de promedio móvil doble es mej
or que el método de promedio móvil simple,
# ya que el error cuadrático medio es menor en el método
de promedio móvil doble.



# (b) Si se utiliza un suavizamiento exponencial simple con una constante
de suavizamiento de 0.4,
# ¿Cuál es el pronóstico para el IV trimestre del 2020?

Yses <- ses(Yts, alpha = 0.4, h = 4)
plot(Yts, type = "l", xlab="Trimestre", ylab=" ")
lines(Yses$fitted, type = "l", col = "red")
legend(x = "bottomright", legend = c("Yts", "Yses"), col = c('black', 're
d'), lty = c(1,
1))
```

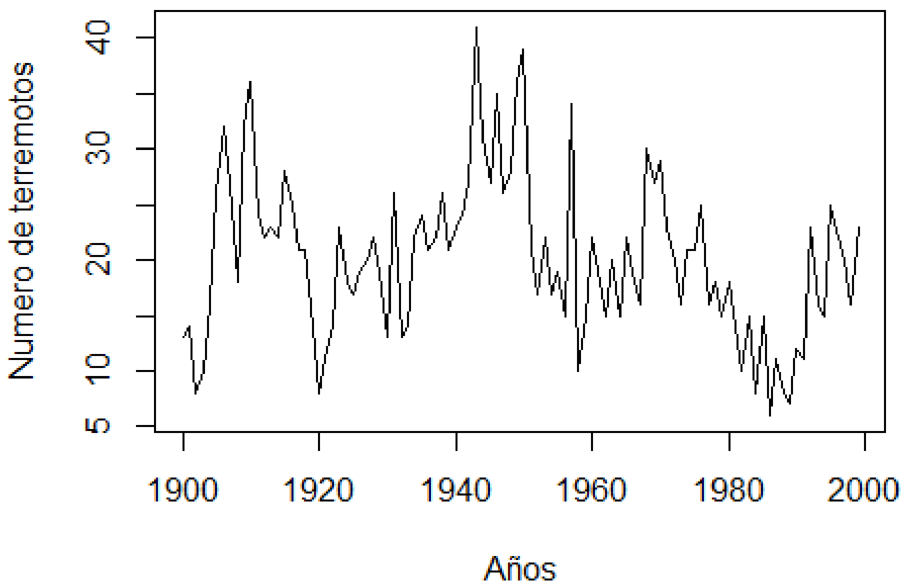



```
Yses$fitted

##          Qtr1     Qtr2     Qtr3     Qtr4
## 2016 267.2871 236.3723 289.8234 293.8940
## 2017 264.3364 270.6018 330.3611 322.2167
## 2018 265.3300 267.1980 304.3188 294.5913
## 2019 252.7548 271.6529 334.9917 316.9950
## 2020 270.1970 290.1182 350.0709 338.0426

# El pronóstico para el IV trimestre del 2020 es 338.0426

# (c) Utilice el método de suavizamiento multiplicativo de Holt-Winters c
on las constantes de
# suavizamiento alpha = beta = gamma = 0.5.

Yhw_m <- HoltWinters(Yts, seasonal = "multiplicative",optim.start = c(0.5
,0.5,0.5))
plot(Yhw_m)
legend("topleft",legend = c("Original","Prediccion"),col = c("black","red
"),lty = c(1,1))
```

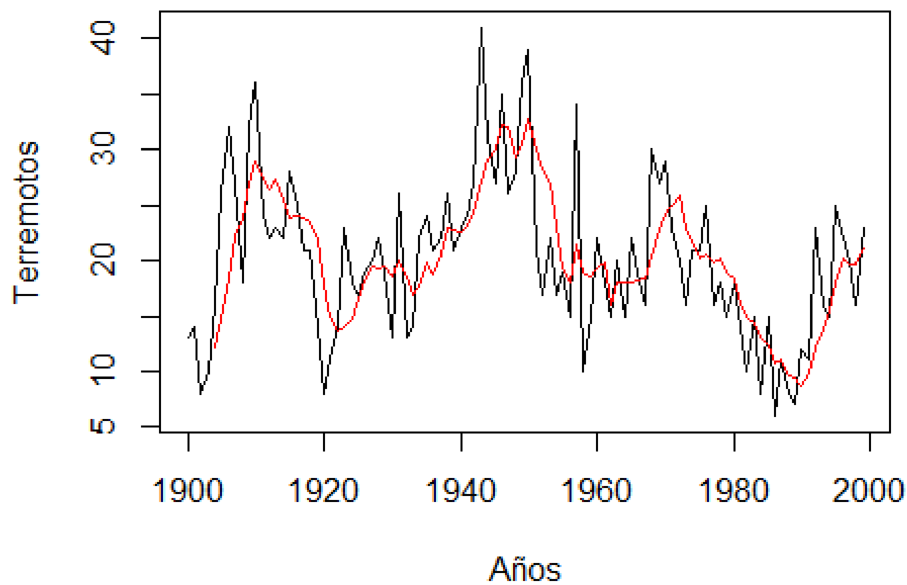



```
# El metodo de Holt Winters se ajusta muy bien a los dato
s originales.

##################################################

#EJERCICIO 5

# La Tabla 5 indica el número de terremotos severos anuales (aquellos con
una magnitud en la escala
# de Richter de 7 grados o más) de 1900 a 1999.

datos = read_excel("F:/777--Programacion repos/Una/r/data/05-t.xlsx")

Yts <- ts(datos$Numero, start = c(1900,1), frequency = 1)
print(Yts)

## Time Series:
## Start = 1900
## End = 1999
## Frequency = 1
##   [1] 13 14  8 10 16 26 32 27 18 32 36 24 22 23 22 28 25 21 21 14  8 1
1 14 23 18
##  [26] 17 19 20 22 19 13 26 13 14 22 24 21 22 26 21 23 24 27 41 31 27 3
5 26 28 36
##  [51] 39 21 17 22 17 19 15 34 10 15 22 18 15 20 15 22 19 16 30 27 29 2
3 20 16 21
```

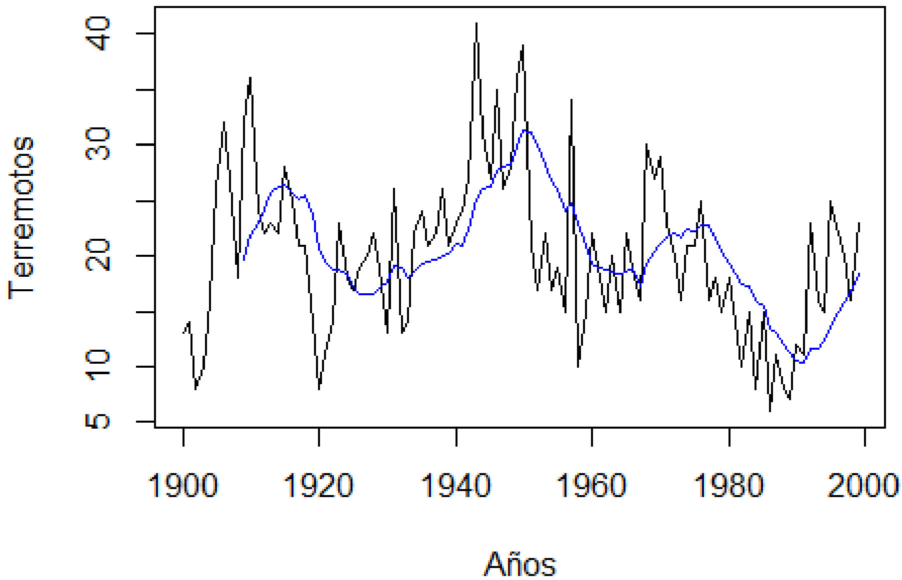
```
##  [76] 21 25 16 18 15 18 14 10 15  8 15  6 11  8  7 12 11 23 16 15 25 2
2 20 16 23

#Graficar la serie temporal
plot(Yts,type = "l", xlab="Años",ylab="Numero de terremotos")
```



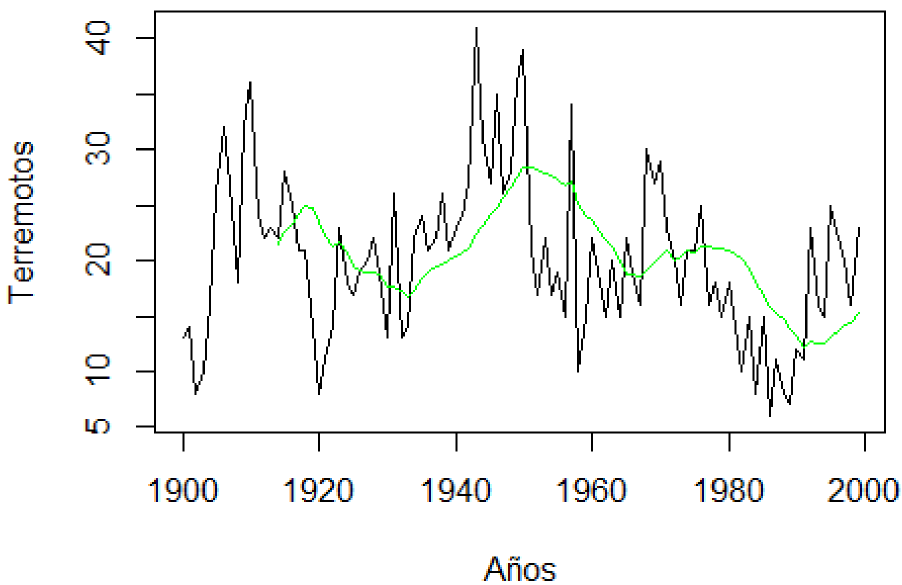

```
# (a) Utilice R-Studio para suavizar los datos de terremotos con promedio
s móviles de órdenes de k
# = 5, 10 y 15. Describa la naturaleza de la suavización conforme el orde
n del promedio móvil se
# incrementa. ¿Cree usted que podría haber un ciclo en estos datos? Si es
así, dé un estimado de
# la duración (en años) del ciclo.

# Promedio móvil simple k = 5
k = 5
Yts_ma <- SMA(Yts,k)

# Graficar la serie temporal con la media movil
plot(Yts, type = "l", xlab = "Años", ylab = "Terremotos") # Serie origina
l
lines(Yts_ma, type = "l", col = "red")
```



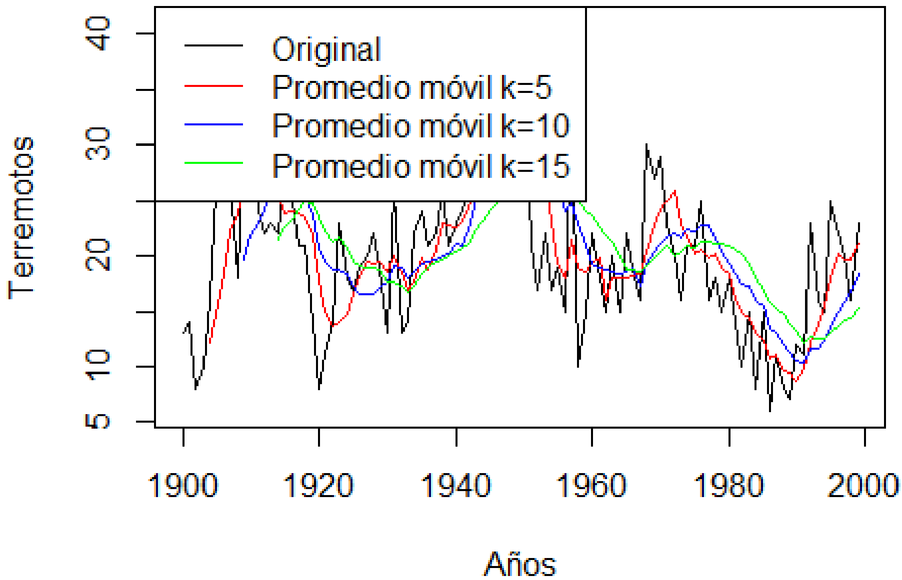

```
# Promedio móvil simple k = 10
k = 10
Yts_ma2 <- SMA(Yts,k)

# Graficar la serie temporal con la media movil
plot(Yts, type = "l", xlab = "Años", ylab = "Terremotos") # Serie origina
l
lines(Yts_ma2, type = "l", col = "blue")
```




```
# Promedio móvil simple k = 15
k = 15
Yts_ma3 <- SMA(Yts,k)

# Graficar la serie temporal con la media movil
plot(Yts, type = "l", xlab = "Años", ylab = "Terremotos") # Serie origina
l
lines(Yts_ma3, type = "l", col = "green")
```

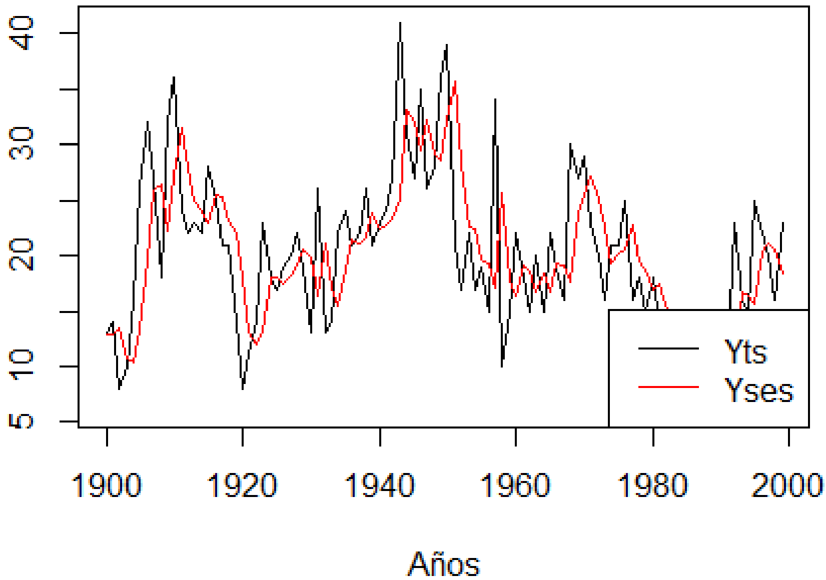



```
# Grafica comparativa entre los 3 promedios móviles
plot(Yts, type = "l", xlab = "Años", ylab = "Terremotos") # Serie origina
l
lines(Yts_ma, type = "l", col = "red")
lines(Yts_ma2, type = "l", col = "blue")
lines(Yts_ma3, type = "l", col = "green")
legend("topleft",legend = c("Original","Promedio móvil k=5","Promedio móv
il k=10","Promedio móvil k=15"),col = c("black","red","blue","green"),lty
= c(1,1))
```



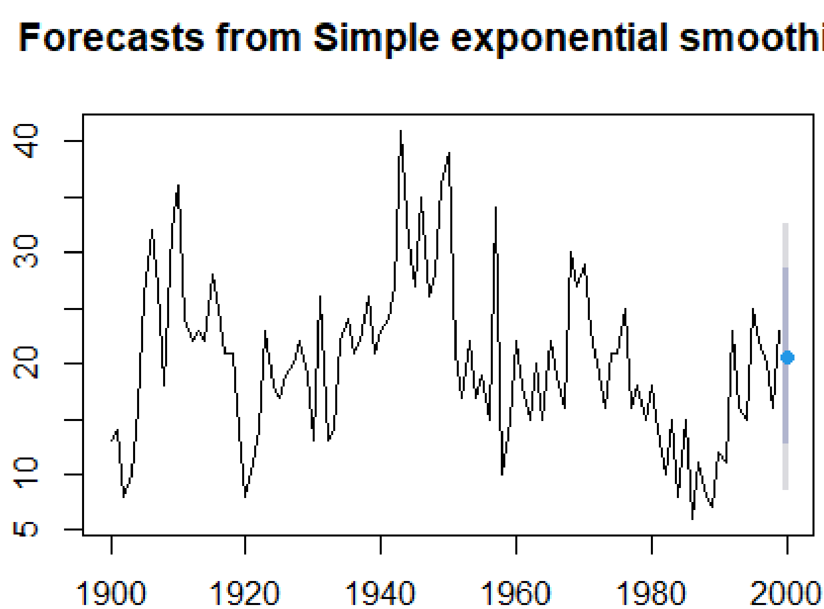

```
# Se observa que conforme el orden del promedio móvil se
incrementa, la suavización de los datos es mayor.
# En este caso, el promedio móvil de orden 15 es el que s
uaviza más los datos. Se observa que hay un ciclo
# en los datos, ya que se observan picos y valles en la s
erie temporal. La duración del ciclo es de
# aproximadamente 50 años.


# (b) Use R-Studio para suavizar los datos de los terremotos usando la su
avización exponencial
# simple. Almacene los residuos y genere un pronóstico para el número de
terremotos severos en
# el año 2000. ¿La suavización exponencial simple ofrece un ajuste razona
ble de estos datos?
# Explique.

# Suavización exponencial simple

Yses <- ses(Yts,alpha = 0.5,h = 1)

plot(Yts,type="l", xlab="Años",ylab=" ")
lines(Yses$fitted, type ="l", col="red")
```

```
legend(x = "bottomright", legend = c("Yts","Yses"),col = c('black','red')
,lty = c(1,1))
```

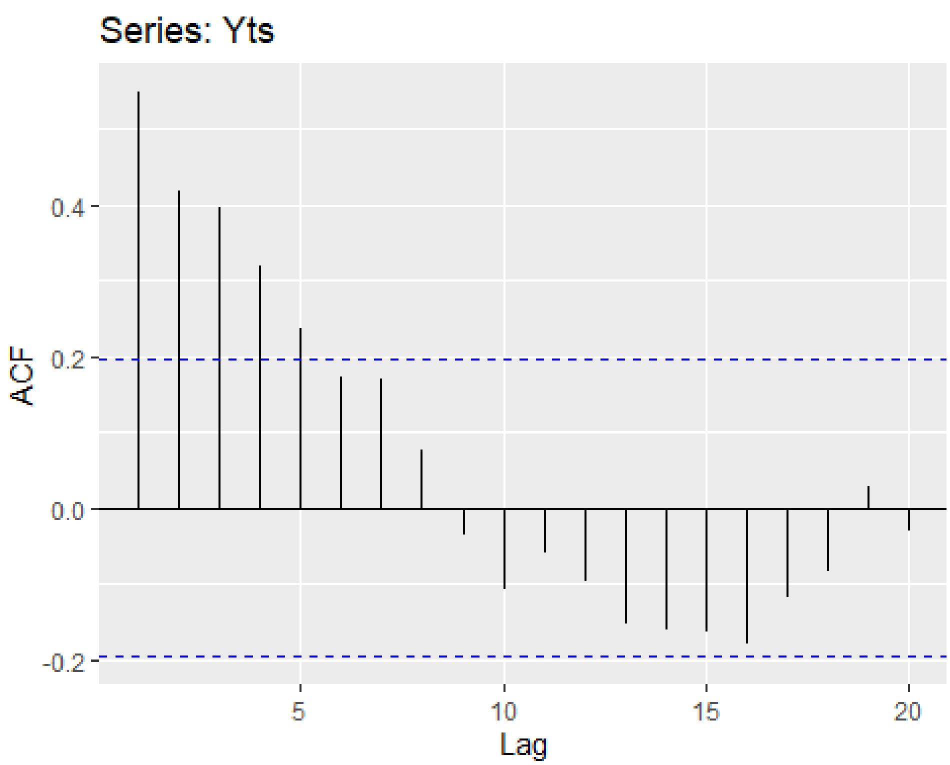



```
# Residuos
residuos <- Yts - Yses$fitted
print(residuos)

## Time Series:
## Start = 1900
## End = 1999
## Frequency = 1
##   [1]   0.015000000   1.007500000  -5.496250000  -0.748125000   5.6259
37500
##   [6]  12.812968750  12.406484375   1.203242187  -8.398378906   9.8008
10547
##  [11]   8.900405273  -7.549797363  -5.774898682  -1.887449341  -1.9437
24670
##  [16]   5.028137665  -0.485931168  -4.242965584  -2.121482792  -8.0607
41396
##  [21] -10.030370698  -2.015185349   1.992407326   9.996203663  -0.0018
98169
##  [26]  -1.000949084   1.499525458   1.749762729   2.874881364  -1.5625
59318
##  [31]  -6.781279659   9.609360171  -8.195319915  -3.097659957   6.4511
70021
##  [36]   5.225585011  -0.387207495   0.806396253   4.403198126  -2.7984
00937
```

```
## [41]   0.600799532   1.300399766   3.650199883  15.825099941  -2.0874
50029
## [46]  -5.043725015   5.478137493  -6.260931254  -1.130465627   7.4347
67187
## [51]   6.717383593 -14.641308203 -11.320654102  -0.660327051  -5.3301
63525
## [56]  -0.665081763  -4.332540881  16.833729559 -15.583135220  -2.7915
67610
## [61]   5.604216195  -1.197891903  -3.598945951   3.200527024  -3.3997
36488
## [66]   5.300131756  -0.349934122  -3.174967061  12.412516470   3.2062
58235
## [71]   3.603129117  -4.198435441  -5.099217721  -6.549608860   1.7251
95570
## [76]   0.862597785   4.431298892  -6.784350554  -1.392175277  -3.6960
87638
## [81]   1.151956181  -3.424021910  -5.712010955   2.143994523  -5.9280
02739
## [86]   4.035998631  -6.982000685   1.508999658  -2.245500171  -2.1227
50086
## [91]   3.938624957   0.969312479  12.484656239  -0.757671880  -1.3788
35940
## [96]   9.310582030   1.655291015  -1.172354493  -4.586177246   4.7069
11377

# Pronósticos
plot(Yses)
```



```r
# El pronóstico para el año 2000 es 20.64654.
# La suavización exponencial simple ofrece un ajuste razonable de los datos, ya que los pronósticos
# se ajustan a la serie temporal. Se observa que los pronósticos siguen la tendencia de los datos
# originales.


# (c) ¿Existe un componente cíclico en los datos del terremoto? ¿Por qué?

ggAcf(Yts)
```




```r
# Se observa que no hay un componente cíclico en los datos del terremoto, ya que no hay picos
# significativos en la función de autocorrelación. Por lo tanto, no hay un patrón cíclico en
# los datos del terremoto.
```