

# Series\_Estacionarias

Arturo

2023-10-31

```
# Instalacion y carga de paquetes
if (!require(zoo)) {
  install.packages("zoo")
}
```

```
## Loading required package: zoo
```

```
##
```

```
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
```

```
##
```

```
##      as.Date, as.Date.numeric
```

```
library(zoo)
```

Nombre: Arturo Garza Campuzano

Matrícula: A00828096

## Series Estacionarias

### Problema #1

#### 1. Datos de las ventas de gasolina en una estación de servicio

```
# Crear un vector para las semanas
```

```
semana <- c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12)
```

```
# Crear un vector para los galones de gasolina (en miles)
```

```
galones_gasolina <- c(17, 21, 19, 23, 18, 16, 20, 18, 22, 20, 15, 22)
```

```
# Combinar los vectores en un data frame
```

```
datos <- data.frame(Semana = semana, Galones_de_gasolina_miles = galones_gasolina)
```

```
# Mostrar el data frame
```

```
print(datos)
```

```
##      Semana Galones_de_gasolina_miles
```

```
## 1         1                        17
```

```
## 2         2                        21
```

```
## 3         3                        19
```

```
## 4         4                        23
```

```
## 5         5                        18
```

```
## 6         6                        16
```

```
## 7      7      20
## 8      8      18
## 9      9      22
## 10     10     20
## 11     11     15
## 12     12     22
```

## 2. Métodos de suavizamiento

```
promedios_moviles <- datos
n <- nrow(promedios_moviles)

promedios_moviles$Pronostico <- NA
promedios_moviles$Error_Pronostico <- NA
promedios_moviles$Error_Cuadrado_Pronostico <- NA
promedios_moviles$Error_Porcentual_Absoluto <- NA

for(i in 1:(n-3)){
  promedios_moviles$Pronostico[i+3]=(promedios_moviles$Galones_de_gasolina_miles[i]+promedios_moviles$Galones_de_gasolina_miles[i+1]+promedios_moviles$Galones_de_gasolina_miles[i+2])/3
  promedios_moviles$Error_Pronostico[i+3]=promedios_moviles$Pronostico[i+3]-promedios_moviles$Galones_de_gasolina_miles[i+3]
  promedios_moviles$Error_Porcentual_Absoluto[i+3]=abs(((promedios_moviles$Galones_de_gasolina_miles[i+3]-promedios_moviles$Pronostico[i+3])/promedios_moviles$Galones_de_gasolina_miles[i+3]))
}

promedios_moviles$Error_Cuadrado_Pronostico <- promedios_moviles$Error_Pronostico^2

# Calcula el cuadrado medio de los errores sin NA:
CME=mean(promedios_moviles$Error_Cuadrado_Pronostico,na.rm=TRUE)

# Calcula el promedio de los errores porcentuales sin NA:
EPAM=mean(promedios_moviles$Error_Porcentual_Absoluto,na.rm=TRUE)

print(promedios_moviles)
```

### 2.1 Promedios móviles

```
##      Semana Galones_de_gasolina_miles Pronostico Error_Pronostico
## 1      1      17      NA      NA
## 2      2      21      NA      NA
## 3      3      19      NA      NA
## 4      4      23      19      -4
## 5      5      18      21      3
## 6      6      16      20      4
## 7      7      20      19      -1
## 8      8      18      18      0
## 9      9      22      18      -4
## 10     10     20      20      0
## 11     11     15      20      5
## 12     12     22      19      -3
##      Error_Cuadrado_Pronostico Error_Porcentual_Absoluto
## 1      NA      NA
## 2      NA      NA
## 3      NA      NA
## 4      16      17.39130
## 5      9      16.66667
```

```
## 6          16          25.00000
## 7           1           5.00000
## 8           0           0.00000
## 9          16          18.18182
## 10          0           0.00000
## 11         25          33.33333
## 12           9          13.63636
```

```
cat("Cuadrado medio de los errores sin NA (CME):",CME,"\n")
```

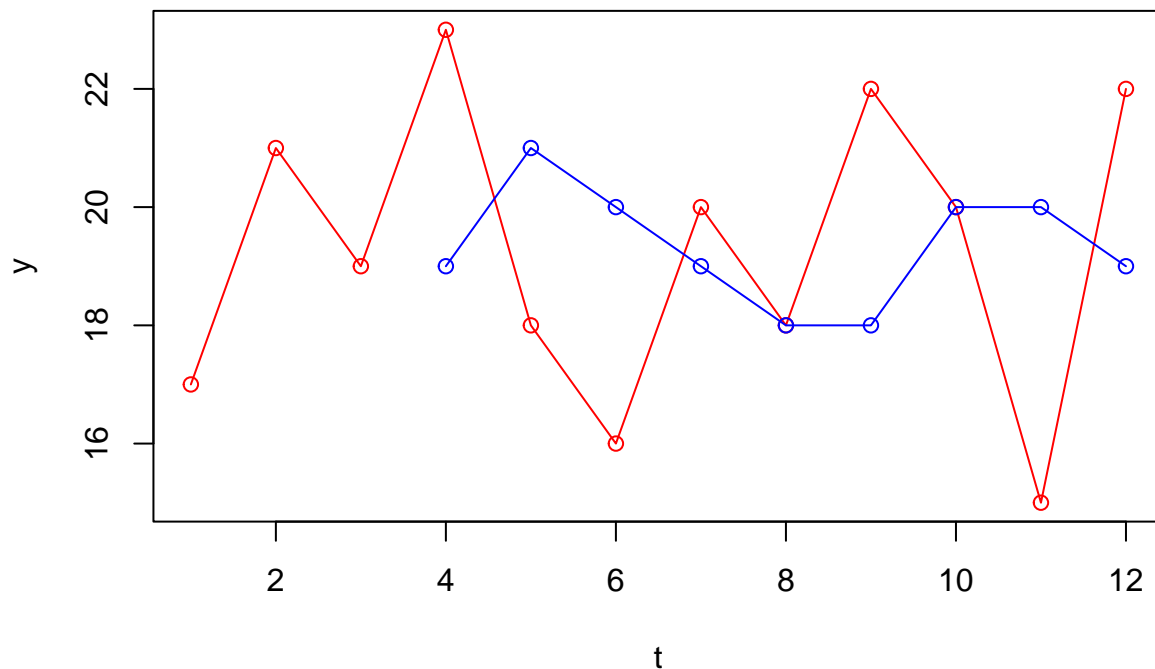
```
## Cuadrado medio de los errores sin NA (CME): 10.22222
```

```
cat("Promedio de los errores porcentuales sin NA (EPAM):",EPAM)
```

```
## Promedio de los errores porcentuales sin NA (EPAM): 14.35661
```

```
resultados <- data.frame(Modelo = "Promedios móviles", CME = CME, EPAM = EPAM)
```

```
plot(promedios_moviles$Semana, promedios_moviles$Galones_de_gasolina_miles, type = "o", col = "red", xlab = "Semana", ylab = "Galones de gasolina (miles)", xlim = (3 + 1):n)
lines(x, promedios_moviles$Pronostico[x], type = "o", col = "blue")
```



```
promedios_moviles_ponderados <- datos
n <- nrow(promedios_moviles_ponderados)
```

```
promedios_moviles_ponderados$Pronostico <- NA
promedios_moviles_ponderados$Error_Pronostico <- NA
promedios_moviles_ponderados$Error_Cuadrado_Pronostico <- NA
promedios_moviles_ponderados$Error_Porcentual_Absoluto <- NA
```

```
for(i in 1:(n-3)){
  promedios_moviles_ponderados$Pronostico[i+3]=((1/6)*promedios_moviles_ponderados$Galones_de_gasolina_miles[i:i+5])
  promedios_moviles_ponderados$Error_Pronostico[i+3]=promedios_moviles_ponderados$Pronostico[i+3]-promedios_moviles_ponderados$Galones_de_gasolina_miles[i+3]
```

```

    promedios_moviles_ponderados$Error_Porcentual_Absoluto[i+3]=abs(((promedios_moviles_ponderados$Galones
})

promedios_moviles_ponderados$Error_Cuadrado_Pronostico <- promedios_moviles_ponderados$Error_Pronostico

# Calcula el cuadrado medio de los errores sin NA:
CME=mean(promedios_moviles_ponderados$Error_Cuadrado_Pronostico,na.rm=TRUE)

# Calcula el promedio de los errores porcentuales sin NA:
EPAM=mean(promedios_moviles_ponderados$Error_Porcentual_Absoluto,na.rm=TRUE)

print(promedios_moviles_ponderados)

```

## 2.2 Promedios móviles ponderados

```

##      Semana Galones_de_gasolina_miles Pronostico Error_Pronostico
## 1         1             17          NA          NA
## 2         2             21          NA          NA
## 3         3             19          NA          NA
## 4         4             23    19.33333    -3.666667
## 5         5             18    21.33333     3.333333
## 6         6             16    19.83333     3.833333
## 7         7             20    17.83333    -2.166667
## 8         8             18    18.33333     0.333333
## 9         9             22    18.33333    -3.666667
## 10        10             20    20.33333     0.333333
## 11        11             15    20.33333     5.333333
## 12        12             22    17.83333    -4.166667
##      Error_Cuadrado_Pronostico Error_Porcentual_Absoluto
## 1                          NA                          NA
## 2                          NA                          NA
## 3                          NA                          NA
## 4          13.44444444          15.942029
## 5          11.1111111          18.518519
## 6          14.6944444          23.958333
## 7           4.6944444          10.833333
## 8           0.1111111           1.851852
## 9          13.4444444          16.666667
## 10          0.1111111           1.666667
## 11          28.4444444          35.555556
## 12          17.3611111          18.939394

```

```

cat("Cuadrado medio de los errores sin NA (CME):",CME,"\n")

```

```

## Cuadrado medio de los errores sin NA (CME): 11.49074

```

```

cat("Promedio de los errores porcentuales sin NA (EPAM):",EPAM)

```

```

## Promedio de los errores porcentuales sin NA (EPAM): 15.99248

```

```

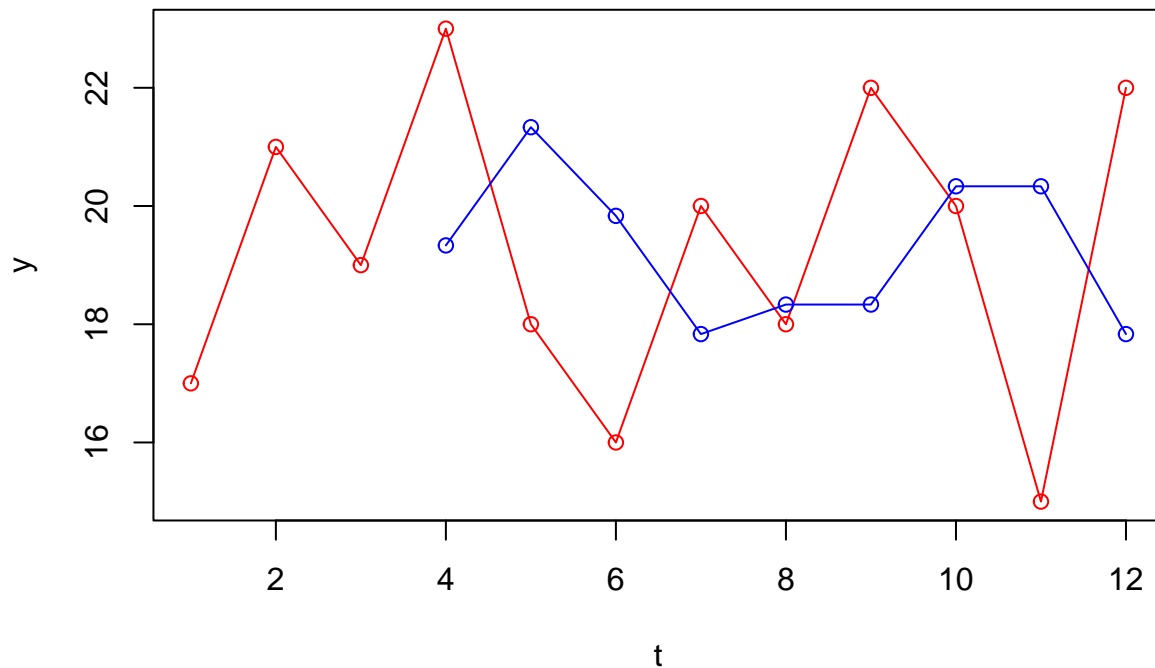
resultados <- rbind(resultados, data.frame(Modelo = "Promedios móviles ponderados", CME = CME, EPAM = EPAM))

```

```

plot(promedios_moviles_ponderados$Semana, promedios_moviles_ponderados$Galones_de_gasolina_miles, type = "n",
x = (3 + 1):n
lines(x, promedios_moviles_ponderados$Pronostico[x], type = "o", col = "blue")

```



```

suavizamiento_exponencial <- datos
n <- nrow(suavizamiento_exponencial)

suavizamiento_exponencial$Pronostico <- NA
suavizamiento_exponencial$Error_Pronostico <- NA
suavizamiento_exponencial$Error_Cuadrado_Pronostico <- NA
suavizamiento_exponencial$Error_Porcentual_Absoluto <- NA

suavizamiento_exponencial$Pronostico[1] <- suavizamiento_exponencial$Galones_de_gasolina_miles[1]
suavizamiento_exponencial$Pronostico[2] <- suavizamiento_exponencial$Galones_de_gasolina_miles[1]
a <- 0.20

for (i in 3:n) {
  suavizamiento_exponencial$Pronostico[i] = (a*suavizamiento_exponencial$Galones_de_gasolina_miles[i-1] +
    (1-a)*suavizamiento_exponencial$Pronostico[i-1])
  suavizamiento_exponencial$Error_Pronostico[i] = suavizamiento_exponencial$Galones_de_gasolina_miles[i] -
    suavizamiento_exponencial$Pronostico[i]
  suavizamiento_exponencial$Error_Cuadrado_Pronostico[i] = suavizamiento_exponencial$Error_Pronostico[i]^2
  suavizamiento_exponencial$Error_Porcentual_Absoluto[i] = abs(suavizamiento_exponencial$Error_Pronostico[i] /
    suavizamiento_exponencial$Galones_de_gasolina_miles[i])
}

suavizamiento_exponencial$Error_Cuadrado_Pronostico <- suavizamiento_exponencial$Error_Pronostico^2

# Calcula el cuadrado medio de los errores sin NA:
CME = mean(suavizamiento_exponencial$Error_Cuadrado_Pronostico, na.rm = TRUE)

# Calcula el promedio de los errores porcentuales sin NA:
EPAM = mean(suavizamiento_exponencial$Error_Porcentual_Absoluto, na.rm = TRUE)

print(suavizamiento_exponencial)

```

### 2.3 Método de suavizamiento exponencial

```

##      Semana Galones_de_gasolina_miles Pronostico Error_Pronostico
## 1      1                          17      17.00000          NA

```

```
## 2      2      21  17.00000      NA
## 3      3      19  17.80000     1.200000
## 4      4      23  18.04000     4.960000
## 5      5      18  19.03200    -1.032000
## 6      6      16  18.82560    -2.825600
## 7      7      20  18.26048     1.739520
## 8      8      18  18.60838    -0.608384
## 9      9      22  18.48671     3.5132928
## 10     10     20  19.18937     0.8106342
## 11     11     15  19.35149    -4.3514926
## 12     12     22  18.48119     3.5188059
```

```
##      Error_Cuadrado_Pronostico Error_Porcentual_Absoluto
```

```
## 1      NA      NA
## 2      NA      NA
## 3      1.4400000      6.315789
## 4      24.6016000      21.565217
## 5      1.0650240      5.733333
## 6      7.9840154      17.660000
## 7      3.0259298      8.697600
## 8      0.3701311      3.379911
## 9      12.3432263      15.969513
## 10     0.6571279      4.053171
## 11     18.9354879      29.009951
## 12     12.3819951      15.994572
```

```
cat("Cuadrado medio de los errores sin NA (CME):",CME,"\n")
```

```
## Cuadrado medio de los errores sin NA (CME): 8.280454
```

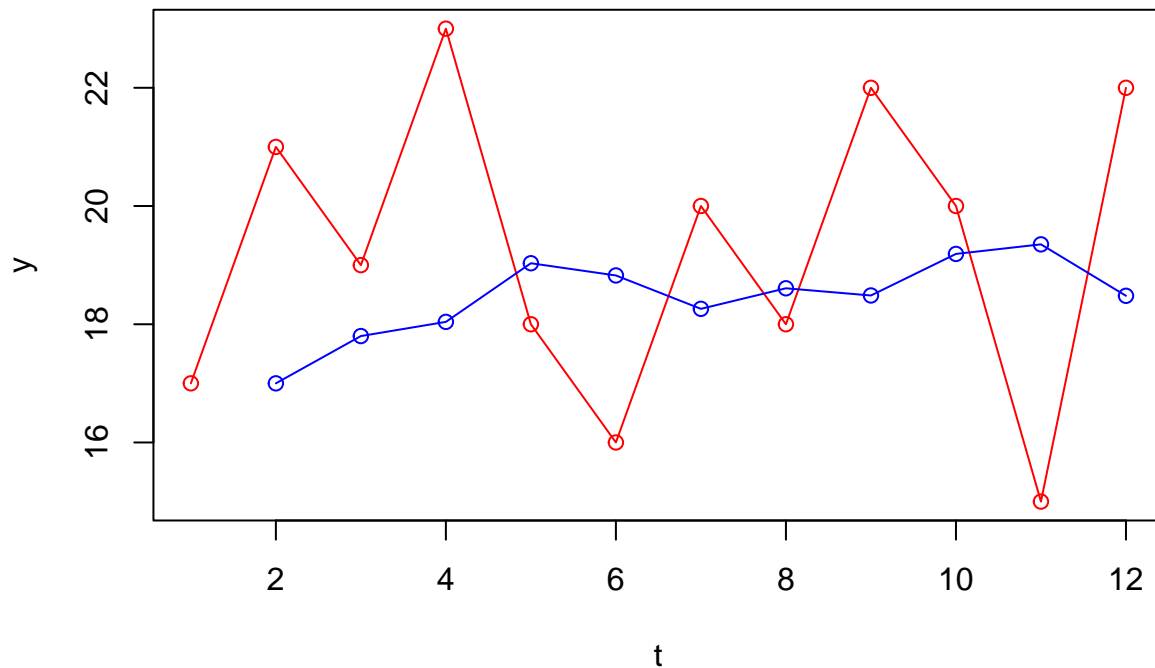
```
cat("Promedio de los errores porcentuales sin NA (EPAM):",EPAM)
```

```
## Promedio de los errores porcentuales sin NA (EPAM): 12.83791
```

```
resultados <- rbind(resultados, data.frame(Modelo = "Suavizamiento exponencial 1", CME = CME, EPAM = EPAM))
```

```
plot(suavizamiento_exponencial$Semana, suavizamiento_exponencial$Galones_de_gasolina_miles, type = "o",
x = 2:n
```

```
lines(x, suavizamiento_exponencial$Pronostico[x], type = "o", col = "blue")
```



```

suavizamiento_exponencial_min <- datos
n <- nrow(suavizamiento_exponencial_min)

suavizamiento_exponencial_min$Pronostico <- NA
suavizamiento_exponencial_min$Error_Pronostico <- NA
suavizamiento_exponencial_min$Error_Cuadrado_Pronostico <- NA
suavizamiento_exponencial_min$Error_Porcentual_Absoluto <- NA

suavizamiento_exponencial_min$Pronostico[1] <- suavizamiento_exponencial_min$Galones_de_gasolina_miles[1]
suavizamiento_exponencial_min$Pronostico[2] <- suavizamiento_exponencial_min$Galones_de_gasolina_miles[2]

# Definir el rango de valores de alpha a probar
alphas <- seq(0, 1, by = 0.01)

# Inicializar variables para almacenar el mejor valor de alpha y el CME mínimo
best_alpha <- 0
min_cme <- Inf

# Bucle para probar diferentes valores de alpha
for (a in alphas) {
  for (i in 3:n) {
    suavizamiento_exponencial_min$Pronostico[i] = (a * suavizamiento_exponencial_min$Galones_de_gasolina_miles[i-1] +
    suavizamiento_exponencial_min$Error_Pronostico[i] = suavizamiento_exponencial_min$Galones_de_gasolina_miles[i] -
    suavizamiento_exponencial_min$Error_Porcentual_Absoluto[i] = abs(((suavizamiento_exponencial_min$Galones_de_gasolina_miles[i] -
  })

suavizamiento_exponencial_min$Error_Cuadrado_Pronostico <- suavizamiento_exponencial_min$Error_Pronostico^2

# Calcula el cuadrado medio de los errores sin NA:
CME = mean(suavizamiento_exponencial_min$Error_Cuadrado_Pronostico, na.rm = TRUE)

```

```

# Verifica si este valor de alpha tiene un CME menor
if (CME < min_cme) {
  min_cme <- CME
  best_alpha <- a
}
}

# Una vez encontrado el mejor valor de alpha, calcula el Pronostico final con ese valor
for (i in 3:n) {
  suavizamiento_exponencial_min$Pronostico[i] = (best_alpha * suavizamiento_exponencial_min$Galones_de_
  suavizamiento_exponencial_min$Error_Pronostico[i] = suavizamiento_exponencial_min$Galones_de_gasolina_
  suavizamiento_exponencial_min$Error_Porcentual_Absoluto[i] = abs(((suavizamiento_exponencial_min$Galones_de_
})

# Calcula el promedio de los errores porcentuales sin NA:
EPAM = mean(suavizamiento_exponencial_min$Error_Porcentual_Absoluto, na.rm = TRUE)

print(suavizamiento_exponencial_min)

```

## 2.4 Valor de $\alpha$ que minimice el CME

```

##      Semana Galones_de_gasolina_miles Pronostico Error_Pronostico
## 1         1              17      17.00000          NA
## 2         2              21      17.00000          NA
## 3         3              19      17.68000      1.3200000
## 4         4              23      17.90440      5.0956000
## 5         5              18      18.77065     -0.7706520
## 6         6              16      18.63964     -2.6396412
## 7         7              20      18.19090      1.8090978
## 8         8              18      18.49845     -0.4984488
## 9         9              22      18.41371      3.5862875
## 10        10              20      19.02338      0.9766186
## 11        11              15      19.18941     -4.1894065
## 12        12              22      18.47721      3.5227926
##      Error_Cuadrado_Pronostico Error_Porcentual_Absoluto
## 1              NA              NA
## 2              NA              NA
## 3              4              6.947368
## 4             16             22.154783
## 5             25              4.281400
## 6              4             16.497757
## 7             16              9.045489
## 8              4              2.769160
## 9             16             16.301307
## 10             4              4.883093
## 11            25             27.929377
## 12            49             16.012694

cat("Alfa que minimiza CME:",best_alpha,"\n")

```

```
## Alfa que minimiza CME: 0.17
```

```
cat("Cuadrado medio de los errores sin NA (CME):",min_cme,"\n")
```

```
## Cuadrado medio de los errores sin NA (CME): 8.256687
```



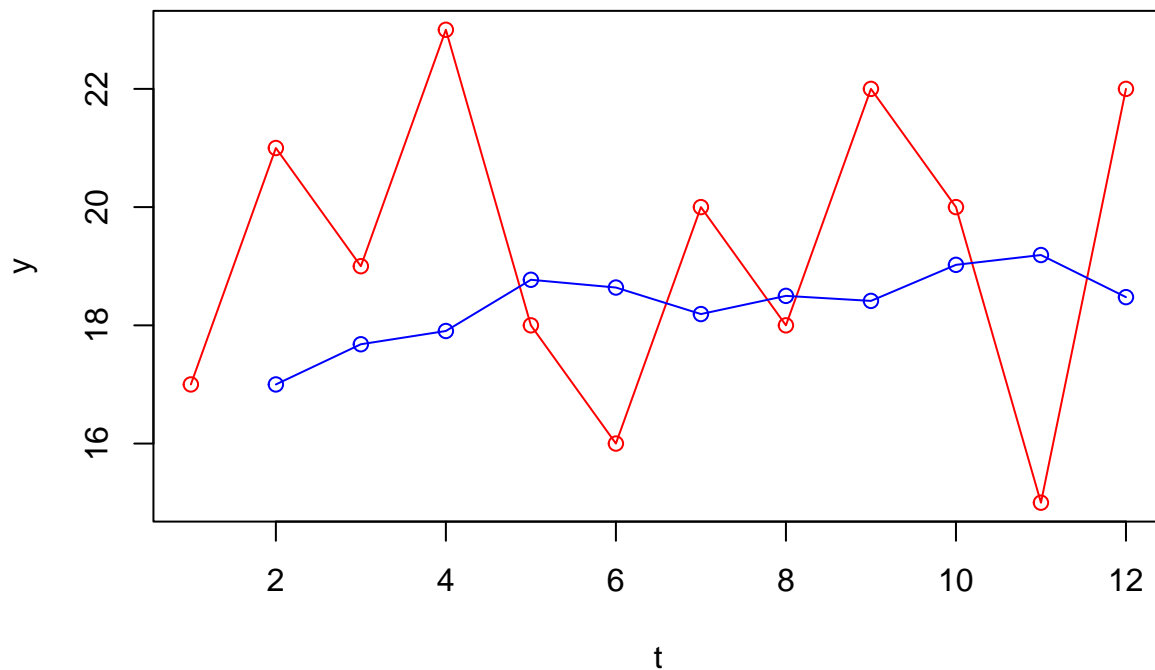
```
cat("Promedio de los errores porcentuales sin NA (EPAM):",EPAM)
```

```
## Promedio de los errores porcentuales sin NA (EPAM): 12.68224
```

```
resultados <- rbind(resultados, data.frame(Modelo = "Suavizamiento exponencial 2", CME = min_cme, EPAM = EPAM))
```

```
plot(suavizamiento_exponencial_min$Semana, suavizamiento_exponencial_min$Galones_de_gasolina_miles, type = "o", col = "red", x = 2:n)
```

```
lines(x, suavizamiento_exponencial_min$Pronostico[x], type = "o", col = "blue")
```



### 3. Conclusiones

```
print(resultados)
```

```
##           Modelo      CME      EPAM
## 1 Promedios móviles 10.222222 14.35661
## 2 Promedios móviles ponderados 11.490741 15.99248
## 3 Suavizamiento exponencial 1 8.280454 12.83791
## 4 Suavizamiento exponencial 2 8.256687 12.68224
```

Bajo el supuesto de que el mejor modelo es aquel que cuente con el menor CME y EPAM posible, el mejor modelo es **Suavizamiento exponencial 2**; con un alfa de 0.17.

### 4. Predicción para la semana 13

```
pred_13 = (best_alpha * suavizamiento_exponencial_min$Galones_de_gasolina_miles[n]) + ((1 - best_alpha) * suavizamiento_exponencial_min$Pronostico[n])
cat("La predicción para la semana 13 es:", pred_13)
```

```
## La predicción para la semana 13 es: 19.07608
```

## Problema #2

Se registró el precio de las acciones de una compañía al cierre de cada día hábil del 24 de agosto al 16 de septiembre. Los datos recopilados son:

### Datos recopilados

```
# Crear un vector para los días
dia <- c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17)

# Crear un vector para los precios
precio <- c(81.32, 81.10, 80.38, 81.34, 80.54, 80.62, 79.54, 79.46, 81.02, 80.98, 80.80, 81.44, 81.48, 80.75, 80.48, 80.01, 80.33)

# Combinar los vectores en un data frame
datos <- data.frame(Dia = dia, Precio = precio)

# Mostrar el data frame
print(datos)
```

```
##      Dia Precio
## 1      1  81.32
## 2      2  81.10
## 3      3  80.38
## 4      4  81.34
## 5      5  80.54
## 6      6  80.62
## 7      7  79.54
## 8      8  79.46
## 9      9  81.02
## 10     10  80.98
## 11     11  80.80
## 12     12  81.44
## 13     13  81.48
## 14     14  80.75
## 15     15  80.48
## 16     16  80.01
## 17     17  80.33
```

a. Use un promedio móvil de tres días para suavizar la serie de tiempo. Pronostique el precio de cierre del 19 de septiembre del 2005 (que es el siguiente día de operaciones).

```
promedios_moviles <- datos
n <- nrow(promedios_moviles)

promedios_moviles$Pronostico <- NA
promedios_moviles$Error_Pronostico <- NA
promedios_moviles$Error_Cuadrado_Pronostico <- NA
promedios_moviles$Error_Porcentual_Absoluto <- NA

for(i in 1:(n-3)){
  promedios_moviles$Pronostico[i+3]=(promedios_moviles$Precio[i]+promedios_moviles$Precio[i+1]+promedios_moviles$Precio[i+2])/3
  promedios_moviles$Error_Pronostico[i+3]=promedios_moviles$Pronostico[i+3]-promedios_moviles$Precio[i+3]
  promedios_moviles$Error_Porcentual_Absoluto[i+3]=abs(((promedios_moviles$Pronostico[i+3]-promedios_moviles$Precio[i+3])/promedios_moviles$Precio[i+3])*100)
}
```

```

promedios_moviles$error_Cuadrado_Pronostico <- promedios_moviles$error_Pronostico^2

# Calcula el cuadrado medio de los errores sin NA:
CME=mean(promedios_moviles$error_Cuadrado_Pronostico,na.rm=TRUE)

# Calcula el promedio de los errores porcentuales sin NA:
EPAM=mean(promedios_moviles$error_Porcentual_Absoluto,na.rm=TRUE)

print(promedios_moviles)

```

```

##      Dia Precio Pronostico Error_Pronostico Error_Cuadrado_Pronostico
## 1      1  81.32          NA              NA              NA
## 2      2  81.10          NA              NA              NA
## 3      3  80.38          NA              NA              NA
## 4      4  81.34    80.93333    -0.40666667    0.165377778
## 5      5  80.54    80.94000     0.40000000    0.160000000
## 6      6  80.62    80.75333     0.13333333    0.017777778
## 7      7  79.54    80.83333     1.29333333    1.672711111
## 8      8  79.46    80.23333     0.77333333    0.598044444
## 9      9  81.02    79.87333    -1.14666667    1.314844444
## 10     10  80.98    80.00667    -0.97333333    0.947377778
## 11     11  80.80    80.48667    -0.31333333    0.098177778
## 12     12  81.44    80.93333    -0.50666667    0.256711111
## 13     13  81.48    81.07333    -0.40666667    0.165377778
## 14     14  80.75    81.24000     0.49000000    0.240100000
## 15     15  80.48    81.22333     0.74333333    0.552544444
## 16     16  80.01    80.90333     0.89333333    0.798044444
## 17     17  80.33    80.41333     0.08333333    0.006944444
##      Error_Porcentual_Absoluto
## 1              NA
## 2              NA
## 3              NA
## 4          0.4999590
## 5          0.4966476
## 6          0.1653849
## 7          1.6260163
## 8          0.9732360
## 9          1.4152884
## 10         1.2019429
## 11         0.3877888
## 12         0.6221349
## 13         0.4991000
## 14         0.6068111
## 15         0.9236249
## 16         1.1165271
## 17         0.1037387

```

```

cat("Cuadrado medio de los errores sin NA (CME):",CME,"\n")

```

```

## Cuadrado medio de los errores sin NA (CME): 0.4995738

```

```

cat("Promedio de los errores porcentuales sin NA (EPAM):",EPAM,"\n")

```

```

## Promedio de los errores porcentuales sin NA (EPAM): 0.7598715

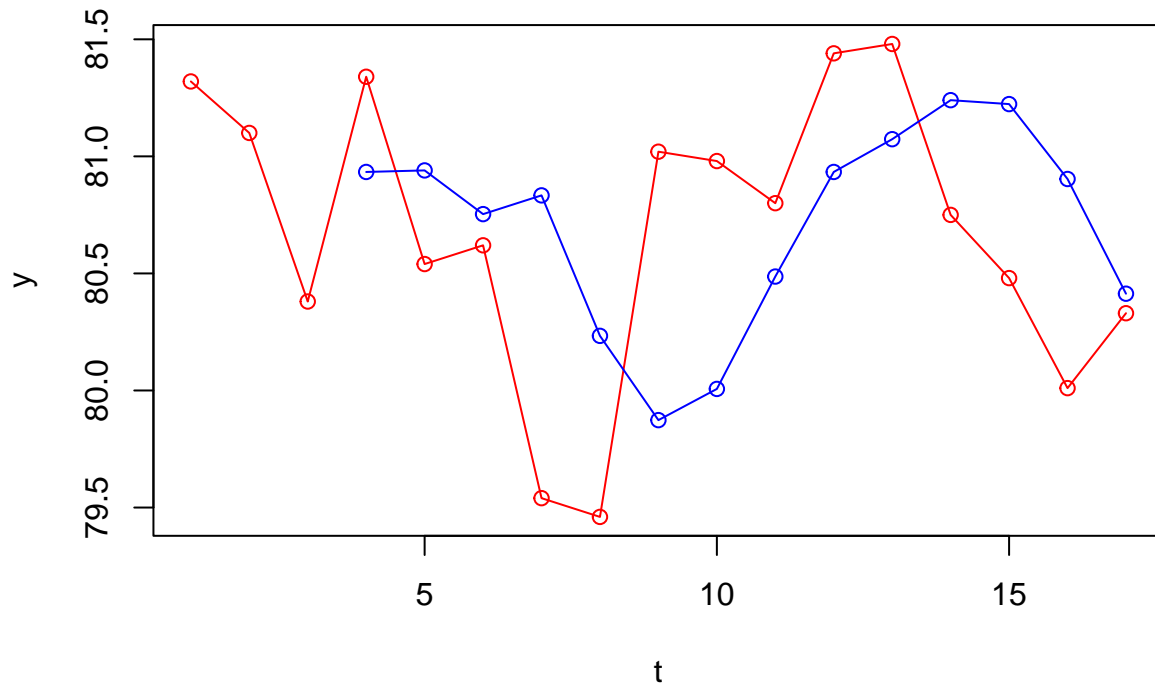
```

```
pronostico_19 = (promedios_moviles$Precio[n-2]+promedios_moviles$Precio[n-1]+promedios_moviles$Precio[n])
cat("El pronóstico del precio de cierre del 19 de septiembre del 2005 es:", pronostico_19)
```

```
## El pronóstico del precio de cierre del 19 de septiembre del 2005 es: 80.27333
```

```
resultados_p2 <- data.frame(Modelo = "Promedios móviles", CME = CME, EPAM = EPAM)
```

```
plot(promedios_moviles$Dia, promedios_moviles$Precio, type = "o", col = "red", xlab = "t", ylab="y")
x = (3 + 1):n
lines(x, promedios_moviles$Pronostico[x], type = "o", col = "blue")
```



b. Emplee el suavizamiento exponencial con  $\alpha = 0.6$  como constante de suavizamiento para suavisar la serie de tiempo. Pronostique el precio de cierre del 19 de septiembre del 2005 (que es el siguiente día de operaciones).

```
suavizamiento_exponencial <- datos
n <- nrow(suavizamiento_exponencial)

suavizamiento_exponencial$Pronostico <- NA
suavizamiento_exponencial$Error_Pronostico <- NA
suavizamiento_exponencial$Error_Cuadrado_Pronostico <- NA
suavizamiento_exponencial$Error_Porcentual_Absoluto <- NA

suavizamiento_exponencial$Pronostico[1] <- suavizamiento_exponencial$Precio[1]
suavizamiento_exponencial$Pronostico[2] <- suavizamiento_exponencial$Precio[1]
a <- 0.6

for (i in 3:n) {
  suavizamiento_exponencial$Pronostico[i] = (a*suavizamiento_exponencial$Precio[i-1]) + ((1-a)*suavizamiento_exponencial$Pronostico[i-1])
  suavizamiento_exponencial$Error_Pronostico[i] = suavizamiento_exponencial$Precio[i] - suavizamiento_exponencial$Pronostico[i]
  suavizamiento_exponencial$Error_Porcentual_Absoluto[i] = abs((suavizamiento_exponencial$Precio[i] - suavizamiento_exponencial$Pronostico[i]) / suavizamiento_exponencial$Precio[i])
}
```

```

suavizamiento_exponencial$Error_Cuadrado_Pronostico <- suavizamiento_exponencial$Error_Pronostico^2

# Calcula el cuadrado medio de los errores sin NA:
CME = mean(suavizamiento_exponencial$Error_Cuadrado_Pronostico, na.rm = TRUE)

# Calcula el promedio de los errores porcentuales sin NA:
EPAM = mean(suavizamiento_exponencial$Error_Porcentual_Absoluto, na.rm = TRUE)

print(suavizamiento_exponencial)

```

```

##      Dia Precio Pronostico Error_Pronostico Error_Cuadrado_Pronostico
## 1      1  81.32   81.32000             NA              NA
## 2      2  81.10   81.32000             NA              NA
## 3      3  80.38   81.18800        -0.80800000      0.6528640000
## 4      4  81.34   80.70320         0.63680000      0.4055142400
## 5      5  80.54   81.08528        -0.54528000      0.2973302784
## 6      6  80.62   80.75811        -0.13811200      0.0190749245
## 7      7  79.54   80.67524        -1.13524480      1.2887807559
## 8      8  79.46   79.99410        -0.53409792      0.2852605881
## 9      9  81.02   79.67364         1.34636083      1.8126874899
## 10     10  80.98   80.48146         0.49854433      0.2485464518
## 11     11  80.80   80.78058         0.01941773      0.0003770484
## 12     12  81.44   80.79223         0.64776709      0.4196022071
## 13     13  81.48   81.18089         0.29910684      0.0894649001
## 14     14  80.75   81.36036        -0.61035727      0.3725359910
## 15     15  80.48   80.99414        -0.51414291      0.2643429278
## 16     16  80.01   80.68566        -0.67565716      0.4565126011
## 17     17  80.33   80.28026         0.04973714      0.0024737826
##      Error_Porcentual_Absoluto
## 1              NA
## 2              NA
## 3          1.00522518
## 4          0.78288665
## 5          0.67703005
## 6          0.17131233
## 7          1.42726276
## 8          0.67215948
## 9          1.66176355
## 10         0.61563884
## 11         0.02403185
## 12         0.79539181
## 13         0.36709234
## 14         0.75586039
## 15         0.63884556
## 16         0.84446589
## 17         0.06191602

```

```

cat("Cuadrado medio de los errores sin NA (CME):",CME,"\n")

```

```

## Cuadrado medio de los errores sin NA (CME): 0.4410245

```

```

cat("Promedio de los errores porcentuales sin NA (EPAM):",EPAM,"\n")

```

```

## Promedio de los errores porcentuales sin NA (EPAM): 0.7000588

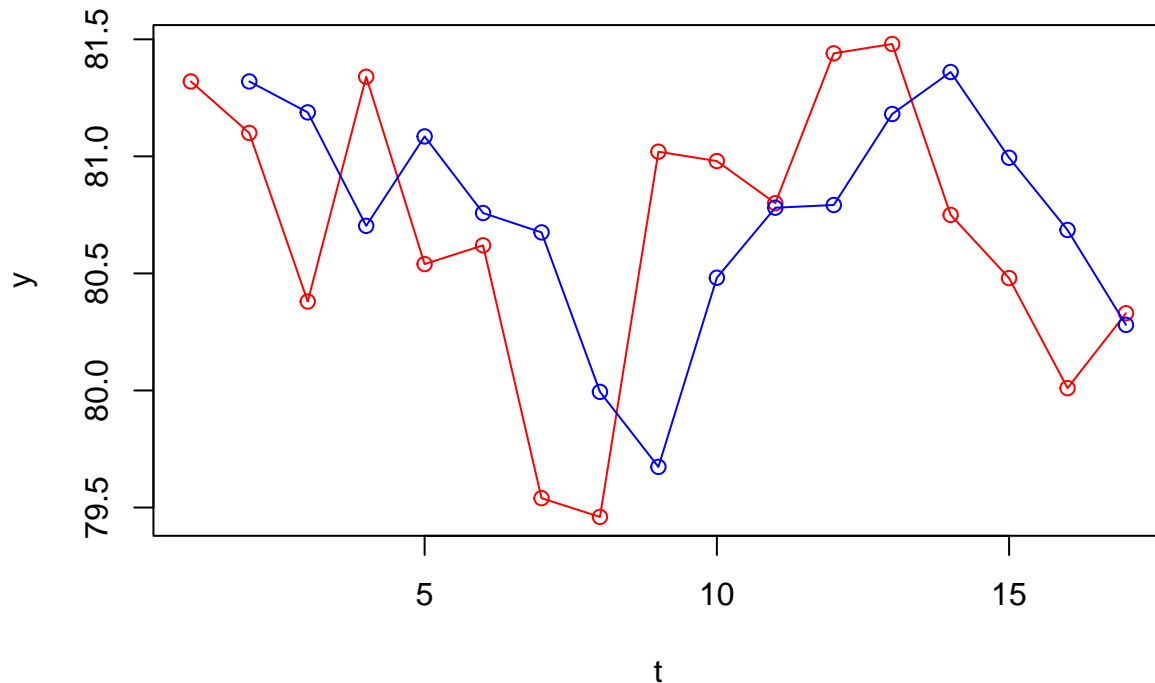
```

```
pronostico_19 = (promedios_moviles$Precio[n-2]+promedios_moviles$Precio[n-1]+promedios_moviles$Precio[n])
cat("El pronóstico del precio de cierre del 19 de septiembre del 2005 es:", pronostico_19)
```

```
## El pronóstico del precio de cierre del 19 de septiembre del 2005 es: 80.27333
```

```
resultados_p2 <- rbind(resultados_p2, data.frame(Modelo = "Suavizamiento exponencial", CME = CME, EPAM = EPAM))
```

```
plot(suavizamiento_exponencial$Dia, suavizamiento_exponencial$Precio, type = "o", col = "red", xlab = "t", ylab = "y")
lines(x = 2:n, suavizamiento_exponencial$Pronostico[x], type = "o", col = "blue")
```



c. ¿Cuál de los dos métodos prefiere? ¿Por qué?

```
print(resultados_p2)
```

```
##           Modelo      CME      EPAM
## 1 Promedios móviles 0.4995738 0.7598715
## 2 Suavizamiento exponencial 0.4410245 0.7000588
```

Aunque el pronóstico de cierre fue el mismo ambos modelos, el mejor modelo es **Suavizamiento exponencial**, ya que es aquel que cuenta con el menor CME y EPAM; con un alfa de 0.6.