

# A7-Introducción a series de tiempo. Series estacionarias

Diego Rodríguez Cantú

2023-11-03

## Problema 1

```
ventas_gasolina = c(17,21,19,23,18,16,20,18,22,20,15,22)
```

### 1. Organización de datos:

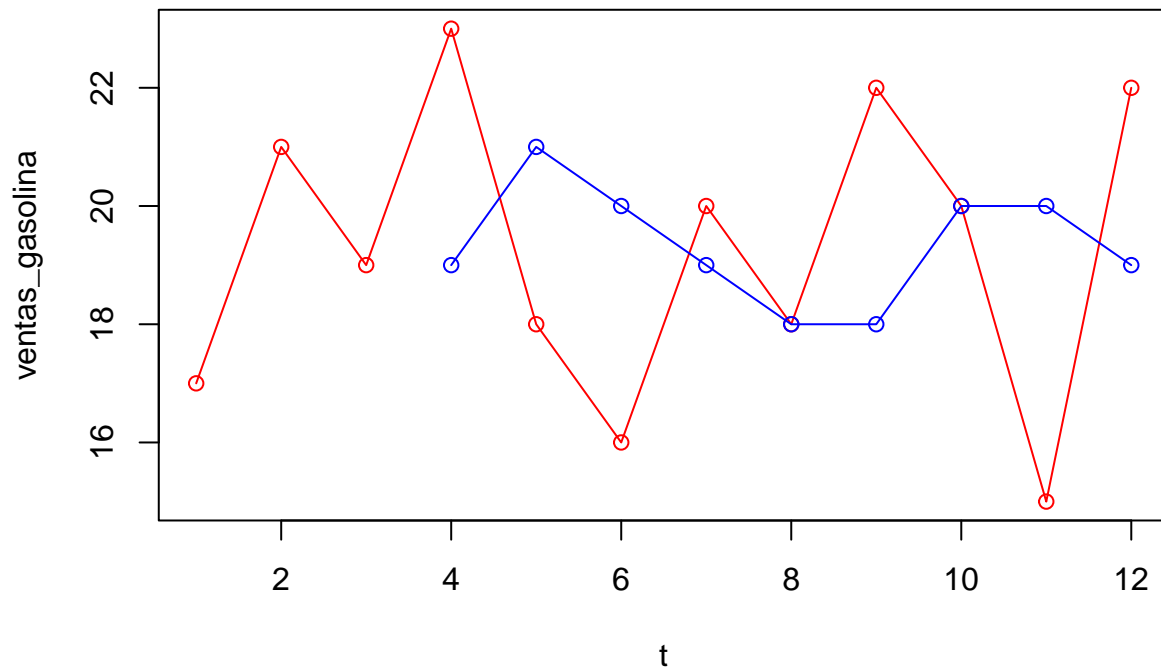
```
t = 1:length(ventas_gasolina)
datos = data.frame(t=t, y=ventas_gasolina)
```

### 2. Cálculo de los pronósticos y errores en promedios móviles:

```
p = rep(NA, length(ventas_gasolina))
e = rep(NA, length(ventas_gasolina))

for(i in 1:(length(ventas_gasolina)-3)) {
  p[i+3] = (ventas_gasolina[i] + ventas_gasolina[i+1] +
            ventas_gasolina[i+2]) / 3
  e[i+3] = p[i+3] - ventas_gasolina[i+3]
}

plot(t, ventas_gasolina, type="o", col="red")
lines(t, p, type="o", col="blue")
```



```
CME = mean(e^2, na.rm=TRUE)
CME
```

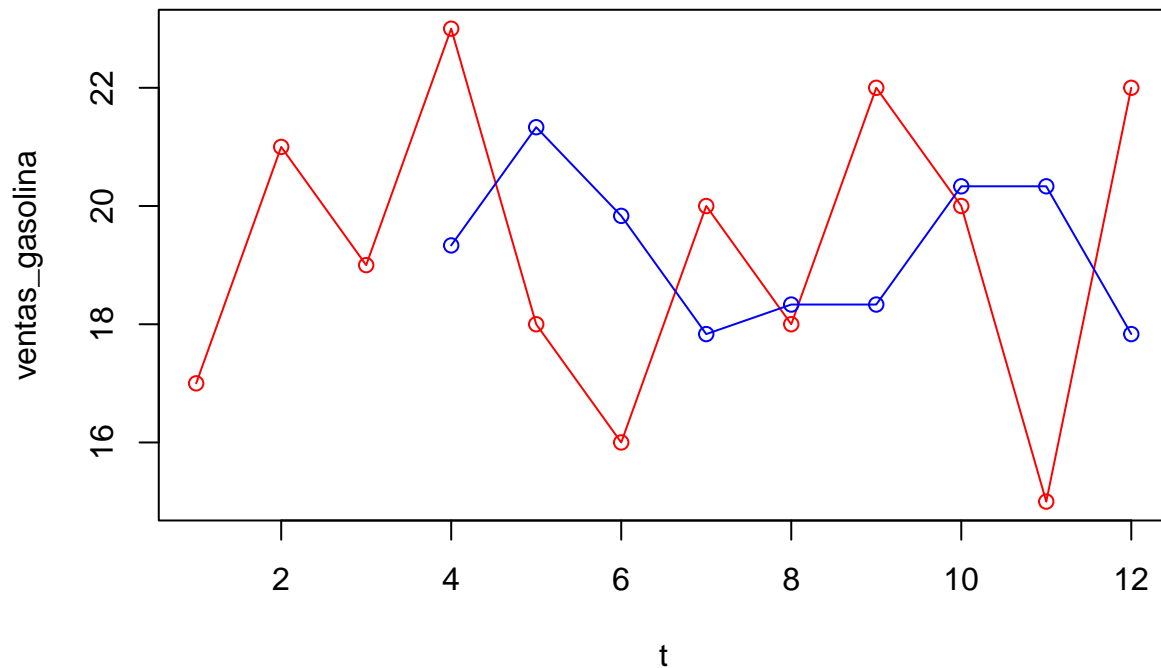
```
## [1] 10.22222
```

### 3. Cálculo de los pronósticos y errores en promedios móviles ponderados:

```
p = rep(NA, length(ventas_gasolina))
e = rep(NA, length(ventas_gasolina))

for(i in 1:(length(ventas_gasolina)-3)) {
  p[i+3] = (1/6)*ventas_gasolina[i] + (2/6)*ventas_gasolina[i+1] +
    (3/6)*ventas_gasolina[i+2]
  e[i+3] = p[i+3] - ventas_gasolina[i+3]
}

plot(t, ventas_gasolina, type="o", col="red")
lines(t, p, type="o", col="blue")
```



```
CME = mean(e^2, na.rm=TRUE)
CME
```

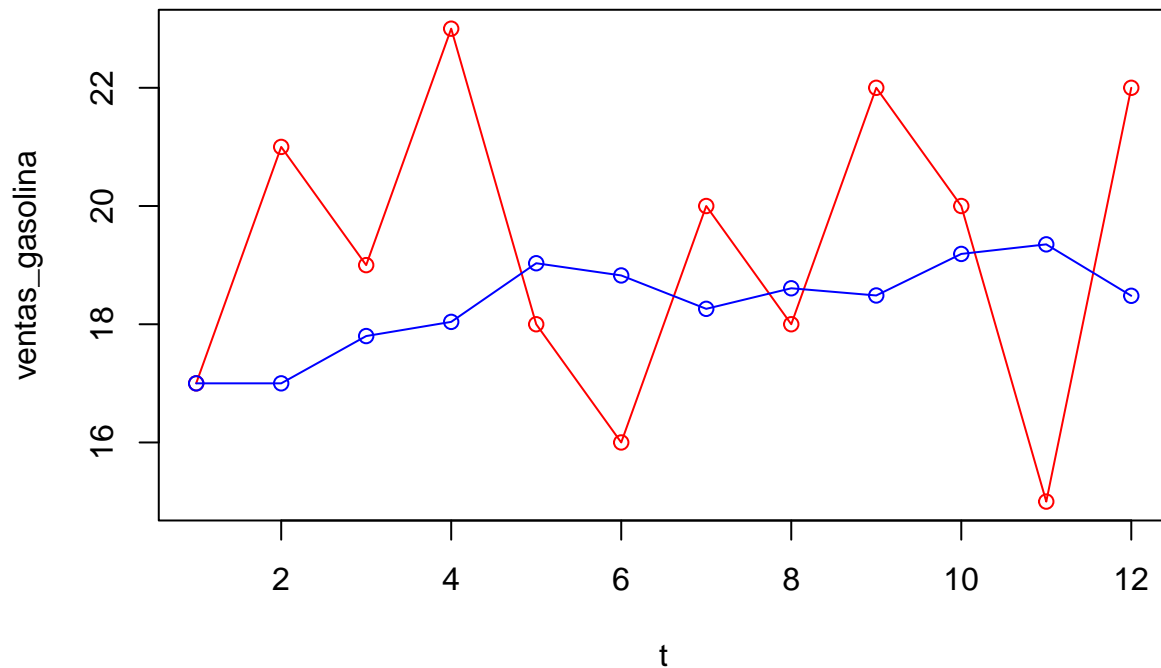
```
## [1] 11.49074
```

#### 4. Cálculo de los pronósticos y errores en suavizamiento exponencial:

```
p = rep(NA, length(ventas_gasolina))
e = rep(NA, length(ventas_gasolina))
p[1] = ventas_gasolina[1]
p[2] = ventas_gasolina[1]
alpha = 0.20

for(i in 3:length(ventas_gasolina)) {
  p[i] = alpha*ventas_gasolina[i-1] + (1-alpha)*p[i-1]
  e[i] = ventas_gasolina[i] - p[i]
}

plot(t, ventas_gasolina, type="o", col="red")
lines(t, p, type="o", col="blue")
```



```
CME = mean(e^2, na.rm=TRUE)
CME
```

```
## [1] 8.280454
```

## 5. Calculo de alpha optimo

```
suavizamiento_exponencial <- function(alpha, datos) {
  n <- length(datos)
  p <- rep(NA, n)
  e <- rep(NA, n)

  p[1] = datos[1]

  for(i in 2:n) {
    p[i] = alpha*datos[i-1] + (1-alpha)*p[i-1]
    e[i] = datos[i] - p[i]
  }

  return(list(p=p, e=e))
}

calcular_CME <- function(e) {
  return(mean(e^2, na.rm=TRUE))
}

alphas = seq(0, 1, by=0.01)
resultados = rep(NA, length(alphas))

for(i in 1:length(alphas)) {
  res = suavizamiento_exponencial(alphas[i], ventas_gasolina)
  resultados[i] = calcular_CME(res$e)
}
```

```

}

alpha_optimo = alphas[which.min(resultados)]

print(paste("El valor óptimo de alpha que minimiza el CME es:",
            round(alpha_optimo, 2)))

```

```
## [1] "El valor óptimo de alpha que minimiza el CME es: 0.17"
```

**Prueba con aplha 0.17**

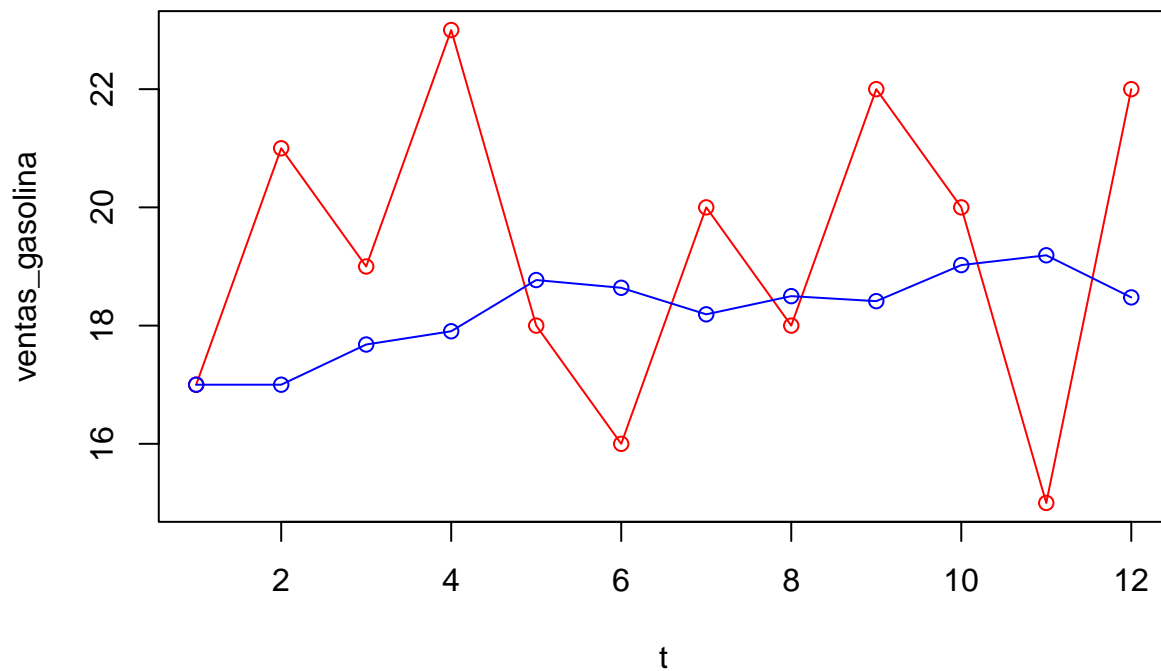
```

p = rep(NA, length(ventas_gasolina))
e = rep(NA, length(ventas_gasolina))
p[1] = ventas_gasolina[1]
p[2] = ventas_gasolina[1]
alpha = 0.17

for(i in 3:length(ventas_gasolina)) {
  p[i] = alpha*ventas_gasolina[i-1] + (1-alpha)*p[i-1]
  e[i] = ventas_gasolina[i] - p[i]
}

plot(t, ventas_gasolina, type="o", col="red")
lines(t, p, type="o", col="blue")

```



```

CMEE = mean(e^2, na.rm=TRUE)
CMEE

```

```
## [1] 8.256687
```

El método con mejores resultados, con un CMEE de 8.25 fue el suavizamiento exponencial con un valor alpha de 0.17.

## 6. Predicción semana 13

```
suavizamiento_exponencial <- function(alpha, datos) {
  n <- length(datos)
  p <- rep(NA, n+1)
  e <- rep(NA, n)

  p[1] = datos[1]

  for(i in 2:n) {
    p[i] = alpha*datos[i-1] + (1-alpha)*p[i-1]
    e[i] = datos[i] - p[i]
  }

  # Predicción para la semana 13
  p[n+1] = alpha*datos[n] + (1-alpha)*p[n]

  return(list(p=p, e=e))
}

res_final = suavizamiento_exponencial(alpha_optimo, ventas_gasolina)
prediccion_semana_13 = res_final$p[13]

print(paste("Las ventas de gasolina esperadas para la semana 13 con el suavizamiento exponencial son:",
## [1] "Las ventas de gasolina esperadas para la semana 13 con el suavizamiento exponencial son: 19.08"
```

## Problema 2

```
t <- c(1:17)
y <- 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)

datos <- data.frame(t, y)

# Pronósticos usando Promedio Móvil de 3 días
p <- rep(NA, length(y))
e <- rep(NA, length(y))
for(i in 1:(length(y)-3)){
  p[i+3] <- (y[i] + y[i+1] + y[i+2])/3
  e[i+3] <- p[i+3] - y[i+3]
}

# Pronósticos usando Promedio Móvil Ponderado
pp <- rep(NA, length(y))
ep <- rep(NA, length(y))
for(i in 1:(length(y)-3)){
  pp[i+3] <- (1/6)*y[i] + (2/6)*y[i+1] + (3/6)*y[i+2]
  ep[i+3] <- pp[i+3] - y[i+3]
}

# Pronósticos usando Suavizamiento Exponencial
alpha <- 0.6
ps <- rep(NA, length(y))
```

```

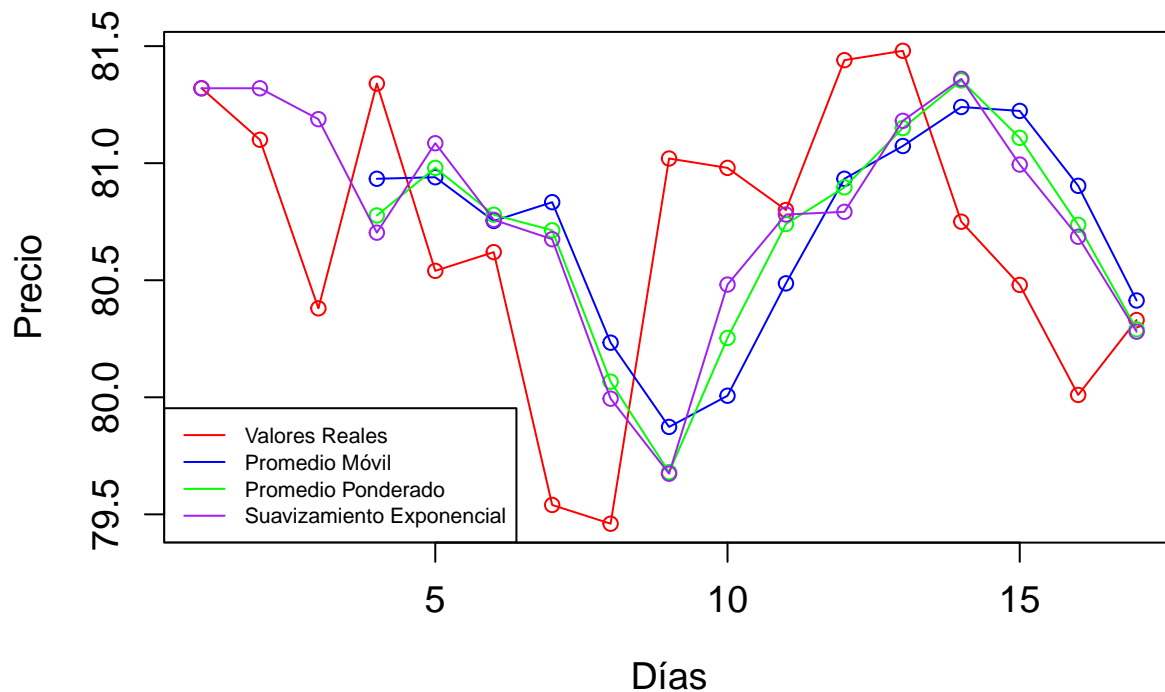
es <- rep(NA, length(y))
ps[1] <- y[1]
for(i in 1:(length(y)-1)){
  ps[i+1] <- alpha*y[i] + (1-alpha)*ps[i]
  es[i+1] <- ps[i+1] - y[i+1]
}

# Calcular el CME para cada método
CME2_pm <- mean(e^2, na.rm=TRUE)
CME2_pp <- mean(ep^2, na.rm=TRUE)
CME2_se <- mean(es^2, na.rm=TRUE)

# Gráficas
plot(t, y, type="o", col="red", main="Pronósticos y Valores Reales",
     xlab="Días", ylab="Precio", cex.lab=1.2, cex.axis=1.2, cex.main=1.4)
lines(t, p, type="o", col="blue")
lines(t, pp, type="o", col="green")
lines(t, ps, type="o", col="purple")
legend("bottomleft", legend=c("Valores Reales", "Promedio Móvil",
                              "Promedio Ponderado", "Suavizamiento Exponencial"),
      col=c("red", "blue", "green", "purple"), lty=1, cex=0.7)

```

## Pronósticos y Valores Reales



```

# Resultados
cat("CME2 Promedio Móvil:", CME2_pm, "\n")

## CME2 Promedio Móvil: 0.4995738

cat("CME2 Promedio Ponderado:", CME2_pp, "\n")

## CME2 Promedio Ponderado: 0.4500796

```

```
cat("CME2 Suavizamiento Exponencial:", CME2_se, "\n")
```

```
## CME2 Suavizamiento Exponencial: 0.4164855
```

### Pronosticos día siguiente

```
# Datos
y <- 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)
```

```
# Promedio móvil de tres días
```

```
p <- NA
e <- NA
for(i in 1:(17-3)){
  p[i+3] <- (y[i] + y[i+1] + y[i+2])/3
  e[i+3] <- p[i+3] - y[i+3]
}
```

```
# Pronóstico para el siguiente día con promedio móvil
```

```
proximo_promedio_movil <- (y[15] + y[16] + y[17])/3
```

```
# Promedio móvil ponderado
```

```
p <- NA
e <- NA
for(i in 1:(17-3)){
  p[i+3] <- (1/6)*y[i] + (2/6)*y[i+1] + (3/6)*y[i+2]
  e[i+3] <- p[i+3] - y[i+3]
}
```

```
# Pronóstico para el siguiente día con promedio móvil ponderado
```

```
proximo_promedio_ponderado <- (1/6)*y[15] + (2/6)*y[16] + (3/6)*y[17]
```

```
# Suavizamiento exponencial
```

```
p <- NA
e <- NA
alpha = 0.6
p[1] = y[1]
for(i in 2:17){
  p[i] <- alpha*y[i-1] + (1-alpha)*p[i-1]
  e[i] <- p[i] - y[i]
}
```

```
# Pronóstico para el siguiente día con suavizamiento exponencial
```

```
proximo_suavizamiento <- alpha*y[17] + (1-alpha)*p[17]
```

```
# Mostrar los pronósticos
```

```
print(paste("Pronóstico para el siguiente día con promedio móvil:",
            proximo_promedio_movil))
```

```
## [1] "Pronóstico para el siguiente día con promedio móvil: 80.2733333333333"
```

```
print(paste("Pronóstico para el siguiente día con promedio móvil ponderado:",
            proximo_promedio_ponderado))
```

```
## [1] "Pronóstico para el siguiente día con promedio móvil ponderado: 80.2483333333333"
```



```
print(paste("Pronóstico para el siguiente día con suavizamiento exponencial:",  
            proximo_suavizamiento))
```

```
## [1] "Pronóstico para el siguiente día con suavizamiento exponencial: 80.3101051459861"
```