

Series_Tiempo

Jorge Юрий A01749489

10/31/2023

#EJERCICIO 1

```
# Crear un vector para la semana
semana <- c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12)

# Crear un vector para los galones de gasolina
galones_gasolina <- c(17, 21, 19, 23, 18, 16, 20, 18, 22, 20, 15, 22)

# Crear un marco de datos
datos_gasolina <- data.frame(Semana = semana, Galones_de_gasolina = galones_gasolina)

# Mostrar el marco de datos
print(datos_gasolina)
```

```
##      Semana Galones_de_gasolina
## 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
```

```

# Datos de ventas de gasolina
semana <- c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12)
x = semana
galones_gasolina <- c(17, 21, 19, 23, 18, 16, 20, 18, 22, 20, 15, 22)
y = galones_gasolina

# Número de observaciones
n <- length(galones_gasolina)

# Inicializa los vectores de pronóstico (p) y errores (e)
p <- rep(NA, n)
e <- rep(NA, n)

# Calcula pronósticos y errores en promedios móviles
for (i in 1:(n-3)) {
  p[i+3] = (y[i] + y[i+1] + y[i+2]) / 3
  e[i+3] = p[i+3] - y[i+3]
}

# Calcula el CME sin NA
CME <- mean(e[1:n]^2, na.rm = TRUE)

# Muestra el CME
print("CME de promedios móviles:")

```

```
## [1] "CME de promedios móviles:"
```

```
print(CME)
```

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

```

# Número de observaciones
n <- length(galones_gasolina)

# Inicializa los vectores de pronóstico (p) y errores (e)
p <- rep(NA, n)
e <- rep(NA, n)

# Calcula pronósticos y errores en promedios móviles
for (i in 1:(n-3)) {
  p[i+3] = (y[i] + y[i+1] + y[i+2]) / 3
  e[i+3] = p[i+3] - y[i+3]
}

# Crea un marco de datos
datos_resultado <- data.frame(Semana = semana, Ventas_reales = galones_gasolina, Pronostico = p, Error_Cuadrado = e^2)

# Calcula el CME sin NA
CME <- mean(e^2, na.rm = TRUE)

# Muestra el CME
print("CME de promedios móviles:")

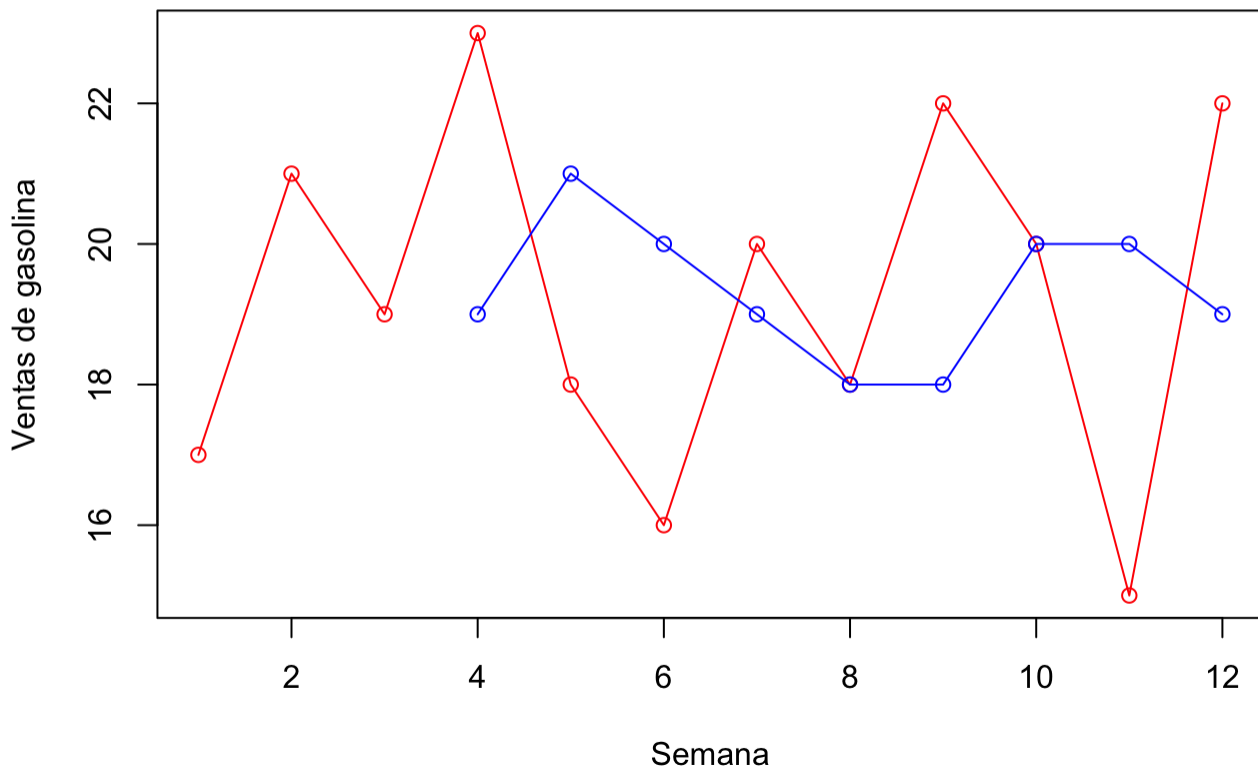
```

```
## [1] "CME de promedios móviles:"
```

```
print(CME)
```

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

```
# Gráfica de las ventas reales y los pronósticos  
plot(semana, galones_gasolina, type = "o", col = "red", xlab = "Semana", ylab = "Ventas de gasolina")  
x <- (3 + 1):n  
lines(x, p[x], type = "o", col = "blue")
```



```

# Inicializa los vectores de pronóstico (p) y errores (e)
p_pmp <- NA
e_pmp <- NA

# Calcula pronósticos y errores en promedios móviles ponderados
for (i in 4:length(galones_gasolina)) {
  p_pmp[i] = (1/6) * galones_gasolina[i-3] + (2/6) * galones_gasolina[i-2] + (3/6) * galones_gasolina[i-1]
  e_pmp[i] = p_pmp[i] - galones_gasolina[i]
}

# Calcula el CME sin NA
CME_pmp <- mean(e_pmp[4:length(e_pmp)]^2, na.rm = TRUE)

# Muestra el CME de promedios móviles ponderados
print("CME de promedios móviles ponderados:")

```

```
## [1] "CME de promedios móviles ponderados:"
```

```
print(CME_pmp)
```

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

```

p_exponencial = numeric(length(y))
e_exponencial = numeric(length(y))
mejor_cme = Inf
for (a in seq(0.01, 0.99, by = 0.01)) {
  for (i in 3:length(galones_gasolina)) {
    p_exponencial[i] <- a * galones_gasolina[i-1] + (1 - a) * p_exponencial[i-1]
    e_exponencial[i] <- galones_gasolina[i] - p_exponencial[i]
  }
  cme <- mean(e_exponencial[3:length(e_exponencial)]^2, na.rm = TRUE)
  if (cme < mejor_cme) {
    mejor_cme <- cme
    mejor_alpha <- a
  }
}

# Muestra el valor de alpha que minimiza el CME
print("Mejor valor de alpha para suavizamiento exponencial:")

```

```
## [1] "Mejor valor de alpha para suavizamiento exponencial:"
```

```
print(mejor_alpha)
```

```
## [1] 0.81
```

```
# Muestra el CME de suavizamiento exponencial
print("CME de suavizamiento exponencial:")
```

```
## [1] "CME de suavizamiento exponencial:"
```

```
print(mejor_cme)
```

```
## [1] 14.13684
```

```
# Utiliza el modelo de promedios móviles para pronosticar la semana 13
pronostico_semana_13 <- mean(galones_gasolina[10:12]) # Promedio de las últimas 3 semanas
```

```
# Muestra el pronóstico para la semana 13
print("Pronóstico de ventas de gasolina para la semana 13 (promedios móviles):")
```

```
## [1] "Pronóstico de ventas de gasolina para la semana 13 (promedios móviles):"
```

```
print(pronostico_semana_13)
```

```
## [1] 19
```

```
# Crear los vectores de fechas y precios
fechas <- as.Date(c("2023-08-24", "2023-08-25", "2023-08-26", "2023-08-29",
                    "2023-08-30", "2023-08-31", "2023-09-01", "2023-09-02",
                    "2023-09-06", "2023-09-07", "2023-09-08", "2023-09-09",
                    "2023-09-12", "2023-09-13", "2023-09-14", "2023-09-15",
                    "2023-09-16"),
                 format = "%Y-%m-%d")

precios <- 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)

base_datos <- data.frame(Dia = fechas, Precio = precios)
```

#Ejercicio 2

```

#EJERCICIO A
y = precios
# Número de observaciones
n <- length(y)

# Inicializa los vectores de pronóstico (p) y errores (e)
p <- rep(NA, n)
e <- rep(NA, n)

# Calcula pronósticos y errores en promedios móviles
for (i in 1:(n-2)) {
  p[i+2] = (y[i] + y[i+1] + y[i+2]) / 3
  if(i+3 <= n) {
    e[i+3] = p[i+2] - y[i+3]
  }
}

# Pronóstico para el siguiente día de operaciones después del último precio disponible
pronostico_siguiente_dia = mean(tail(y, 3))

# Añade el pronóstico al final del vector de pronósticos
p[n] <- pronostico_siguiente_dia

# Calcula el CME sin los NA iniciales
CME <- mean((y[4:n] - p[4:n])^2, na.rm = TRUE)

# Muestra el CME y el pronóstico para el siguiente día de operaciones
print("CME de promedios móviles:")

```

```
## [1] "CME de promedios móviles:"
```

```
print(CME)
```

```
## [1] 0.2108405
```

```
print("Pronóstico para el siguiente día de operaciones:")
```

```
## [1] "Pronóstico para el siguiente día de operaciones:"
```

```
print(pronostico_siguiente_dia)
```

```
## [1] 80.27333
```

#EJERCICIO B

```
# Inicializa los vectores de pronóstico (p) y errores (e)
p_exponencial <- rep(NA, length(precios))
e_exponencial <- rep(NA, length(precios))
alfa <- 0.6

# Inicializa el primer pronóstico
p_exponencial[1] <- precios[1]

# Calcula los pronósticos con suavizamiento exponencial
for (i in 2:length(precios)) {
  p_exponencial[i] <- alfa * precios[i-1] + (1 - alfa) * p_exponencial[i-1]
  e_exponencial[i] <- precios[i] - p_exponencial[i]
}

# Calcula el CME para el suavizamiento exponencial
cme_exponencial <- mean(e_exponencial[2:length(e_exponencial)]^2, na.rm = TRUE)

# Pronóstico para el siguiente día de operaciones después del último precio disponible
pronostico_exponencial_siguiente_dia <- alfa * precios[length(precios)] + (1 - alfa) * p_exponencial[length(precios)]

# Muestra el CME de suavizamiento exponencial
print("CME de suavizamiento exponencial con alfa = 0.6:")
```

```
## [1] "CME de suavizamiento exponencial con alfa = 0.6:"
```

```
print(cme_exponencial)
```

```
## [1] 0.4164855
```

```
# Muestra el pronóstico para el siguiente día de operaciones con suavizamiento exponencial
print("Pronóstico para el siguiente día de operaciones con suavizamiento exponencial:")
```

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

```
print(pronostico_exponencial_siguiente_dia)
```

```
## [1] 80.31011
```

Inciso c: Comparación de métodos de pronóstico

Basándonos en los resultados obtenidos:

- El CME para el promedio móvil fue de 0.2108405.

- El CME para el suavizamiento exponencial con $\alpha = 0.6$ fue de 0.4164855.

Esto sugiere que el promedio móvil proporciona una mejor precisión en este conjunto

de datos, ya que tiene un CME más bajo, indicando errores de pronóstico menores en promedio.

Además, aunque ambos métodos ofrecen pronósticos cercanos para el siguiente día de operaciones:

- Promedio móvil dio 80.27333.

- Suavizamiento exponencial dio 80.31011.

Se prefiere el promedio móvil debido a su mayor precisión histórica evidenciada por su CME más bajo.

Por lo tanto, para este caso particular, el promedio móvil sería el método de pronóstico recomendado.