

## Actividad 8 - Series de tiempo

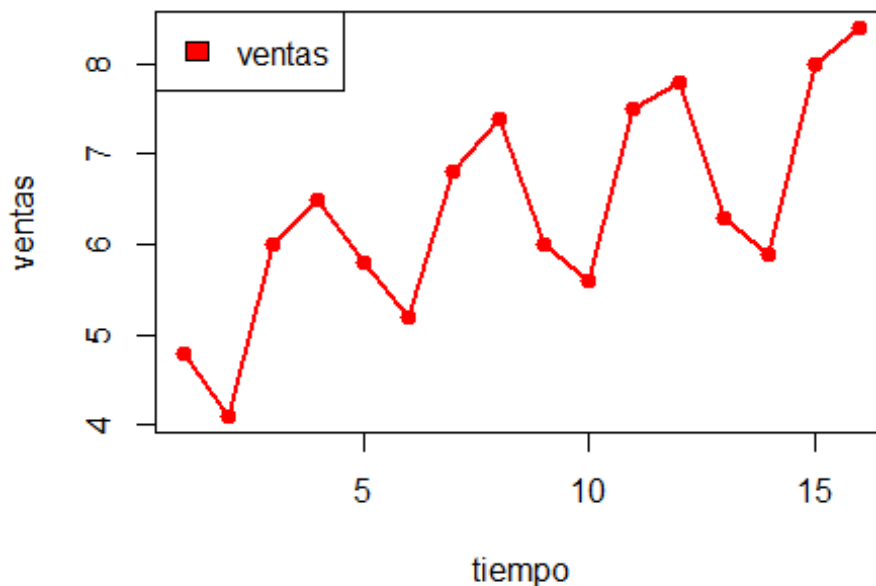
José Carlos Sánchez Gómez

2024-11-13

### Realiza el análisis de tendencia y estacionalidad:

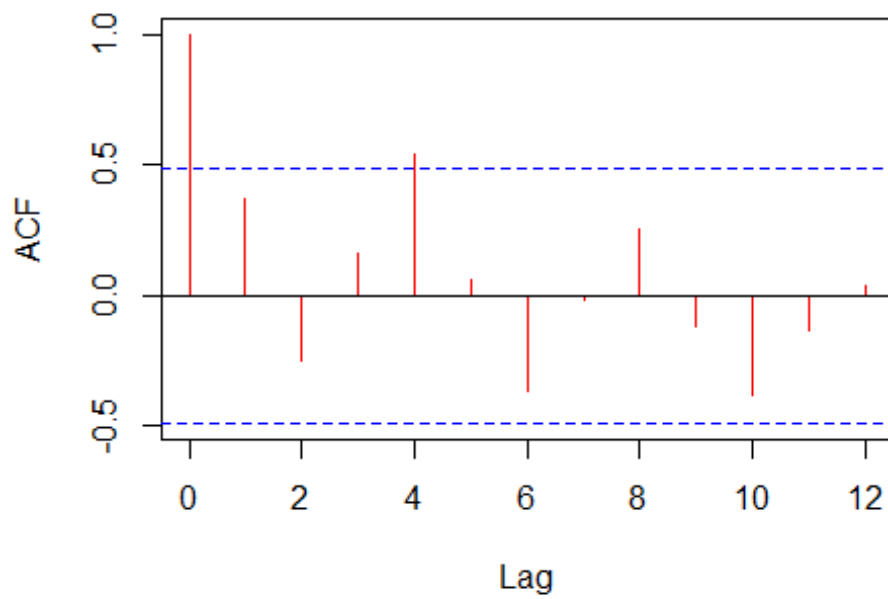
- Identifica si es una serie estacionaria
- Grafica la serie para verificar su tendencia y estacionalidad
- Analiza su gráfico de autocorrelación
- Identifica si el modelo puede ser sumativo o multiplicativo (puedes probar con ambos para ver con cuál es mejor el modelo)

```
ventas = c(4.8, 4.1, 6, 6.5, 5.8, 5.2, 6.8, 7.4, 6, 5.6, 7.5, 7.8, 6.3, 5.9, 8, 8.4)
x= ts(ventas, frequency = 4, start=c(2016,1))
tiempo = 1:16
plot(tiempo, ventas, col = "red", type = "o", lwd = 2, pch = 19)
legend("topleft", legend = c("ventas"), fill = c("red"))
```



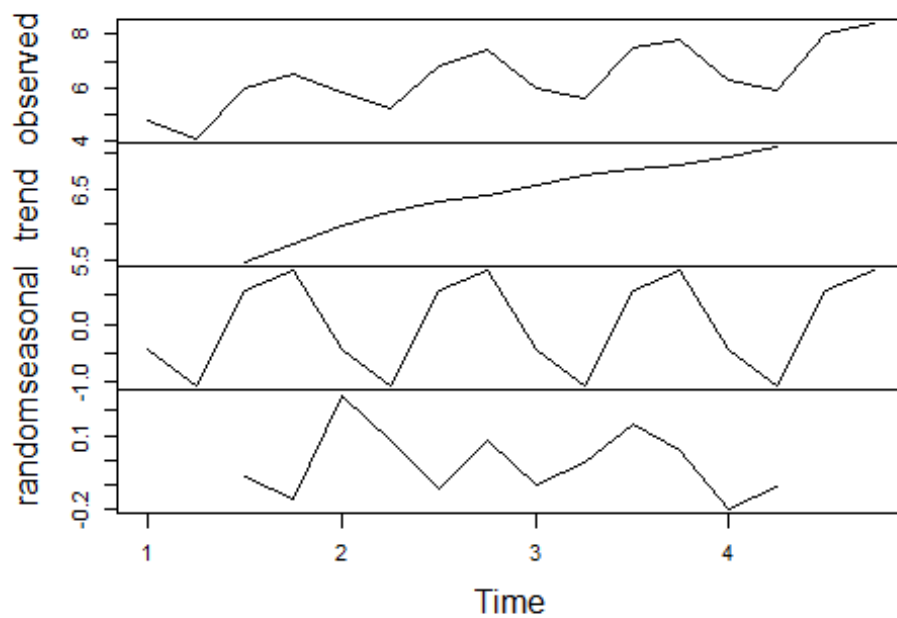
```
ventas_st = ts(ventas, start = 0:12, frequency= 4)
acf(ventas, col = "red", main = "ACF ventas")
```

### ACF ventas



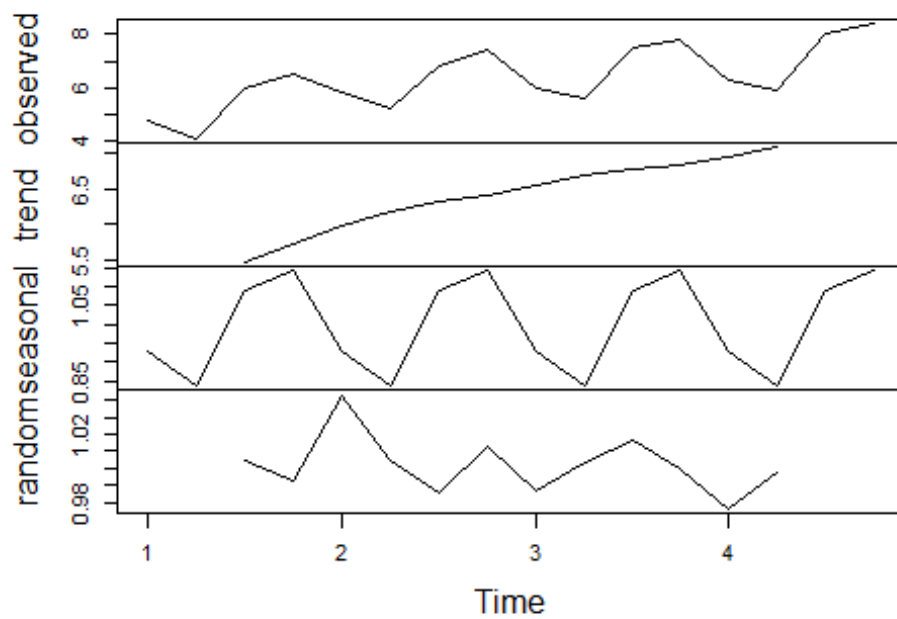
```
qnorm(1-0.05/2)/sqrt(length(ventas))  
## [1] 0.489991  
plot(decompose(x, type = "additive"))
```

## Decomposition of additive time series



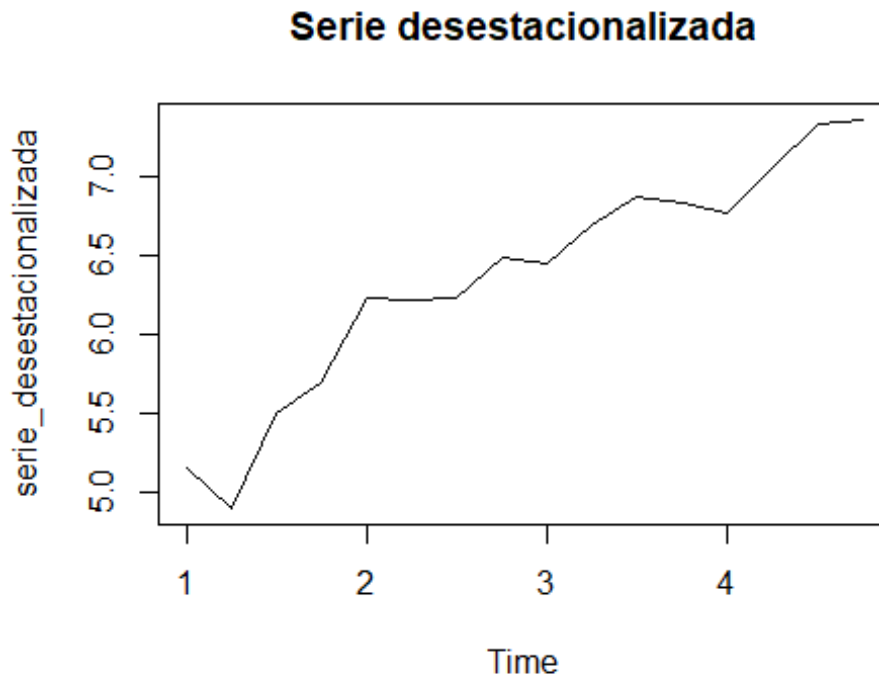
```
plot(decompose(x, type = "multiplicative"))
```

## Decomposition of multiplicative time series



## Calcula los índices estacionales y grafica la serie desestacionalizada

```
indices = decompose(x, type = "multiplicative")
serie_desestacionalizada = x / indices$seasonal
plot(serie_desestacionalizada, main = "Serie desestacionalizada")
```



## Analiza el modelo lineal de la tendencia

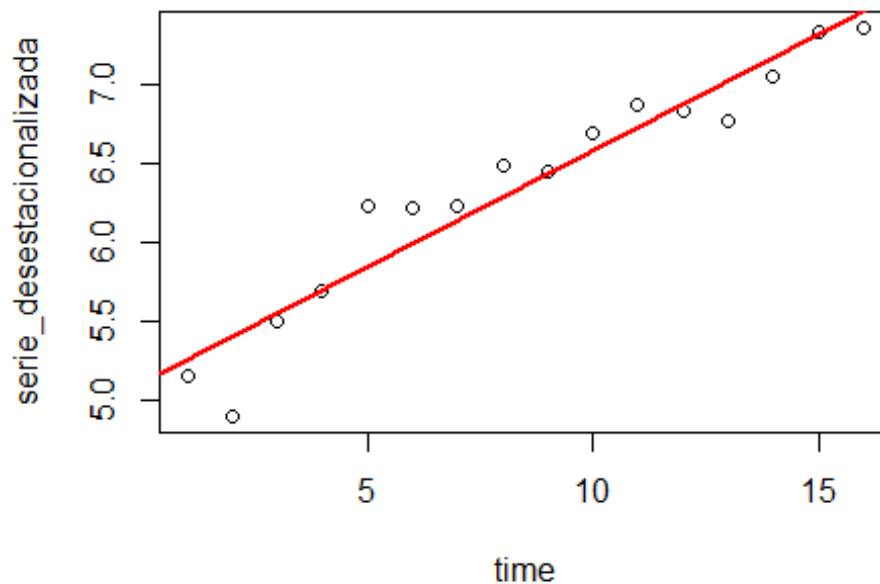
- Realiza la regresión lineal de la tendencia (ventas desestacionalizadas vs tiempo)
- Analiza la significancia del modelo lineal, global e individual
- Haz el análisis de residuos

```
time = 1:length(serie_desestacionalizada)
modelo_lineal = lm(serie_desestacionalizada ~ time)
summary(modelo_lineal)
```

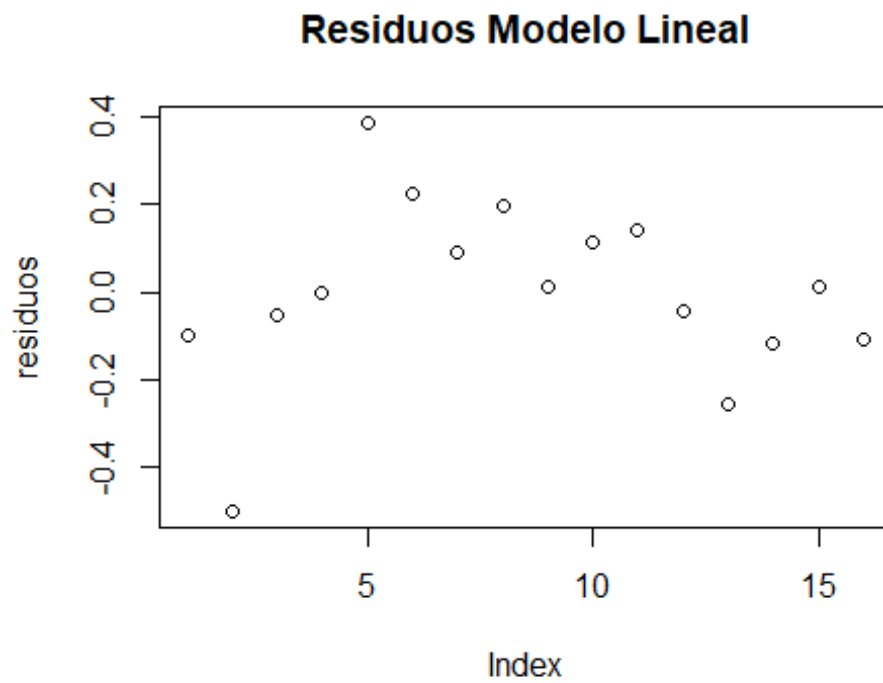
```
##
## Call:
## lm(formula = serie_desestacionalizada ~ time)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.5007 -0.1001  0.0037  0.1207  0.3872
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   5.10804    0.11171   45.73  < 2e-16 ***
```

```
## time          0.14738    0.01155   12.76 4.25e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.213 on 14 degrees of freedom
## Multiple R-squared:  0.9208, Adjusted R-squared:  0.9151
## F-statistic: 162.7 on 1 and 14 DF,  p-value: 4.248e-09

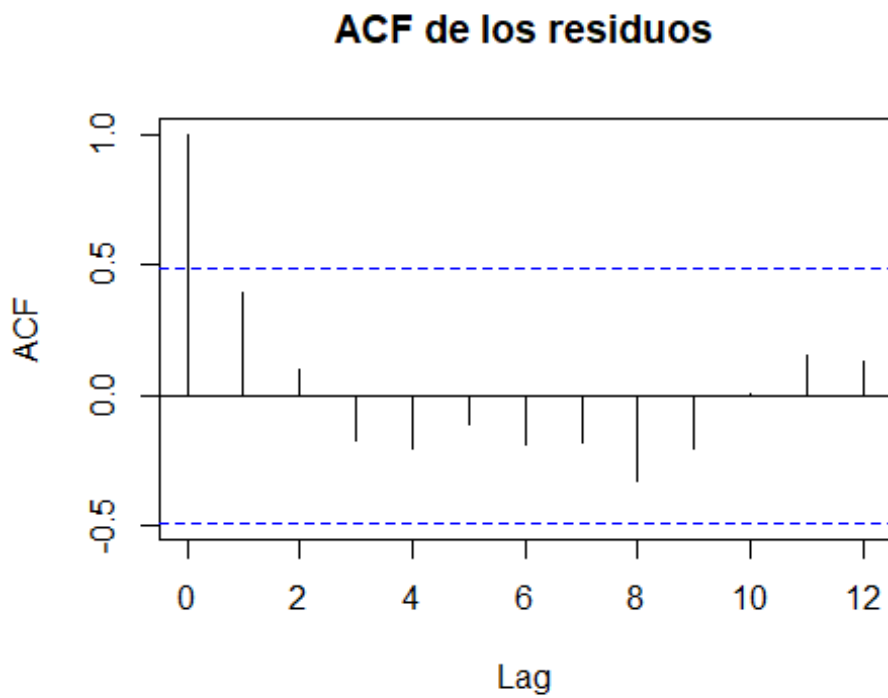
plot(time, serie_desestacionalizada)
abline(modelo_lineal, col = "red", lwd = 2)
```



```
residuos = modelo_lineal$residuals
plot(residuos, main = "Residuos Modelo Lineal")
```



```
acf(residuos, main = "ACF de los residuos")
```



CME y el EPAM de la predicción de la serie de tiempo

## Calcula el

```

cme = mean(residuos ^ 2)
epam = mean(abs(residuos / serie_desestacionalizada)) * 100
cat("CME:", cme, "\n")

## CME: 0.0397064

cat("EPAM:", epam, "\n")

## EPAM: 2.439533

```

Explora un mejor modelo, por ejemplo un modelo cuadrático:  $y = B_0 + B_1x + B_2x^2$ . Para ello transforma la variable ventas (recuerda que la regresión no lineal es una regresión lineal con una transformación).

```

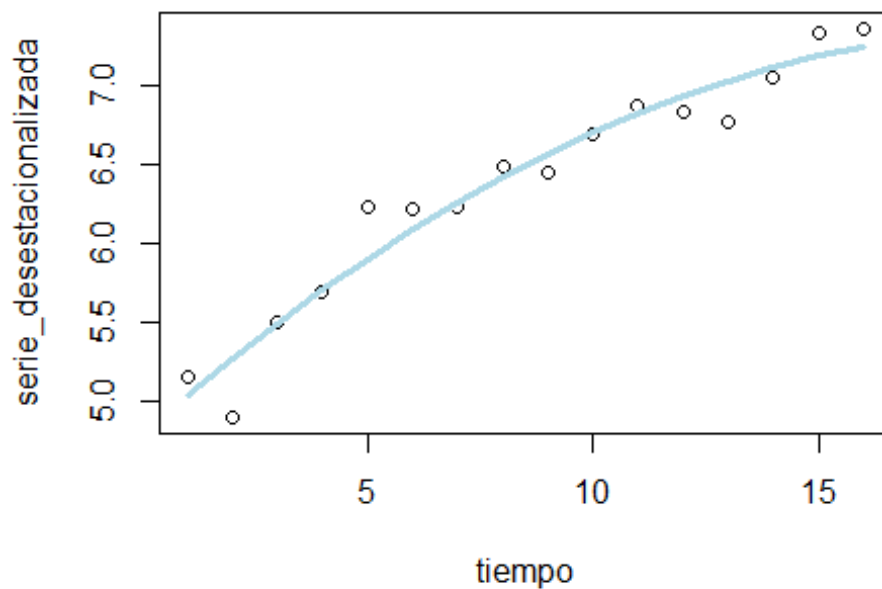
modelo_cuadr = lm(serie_desestacionalizada ~ time + I(time ^2))
summary(modelo_cuadr)

##
## Call:
## lm(formula = serie_desestacionalizada ~ time + I(time^2))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.36986 -0.07058 -0.00100  0.11345  0.33110
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.790283   0.152429  31.426 1.20e-13 ***
## time         0.253302   0.041269   6.138 3.56e-05 ***
## I(time^2)    -0.006231   0.002360  -2.640  0.0204 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1784 on 13 degrees of freedom
## Multiple R-squared:  0.9484, Adjusted R-squared:  0.9405
## F-statistic: 119.6 on 2 and 13 DF,  p-value: 4.268e-09

plot(tiempo, serie_desestacionalizada, main = "Modelo cuadrático de
ventas desestacionalizadas")
lines(tiempo, predict(modelo_cuadr), col = "lightblue", lwd = 3)

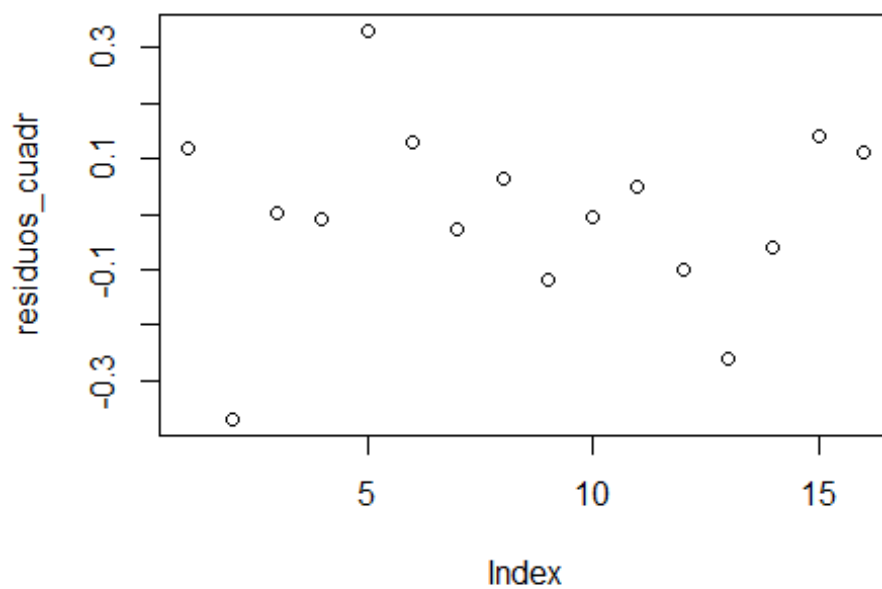
```

## Modelo cuadratico de ventas desestacionalizada



```
residuos_cuadr = modelo_cuadr$residuals  
plot(residuos_cuadr, main = "Residuos Modelo Cuadratico")
```

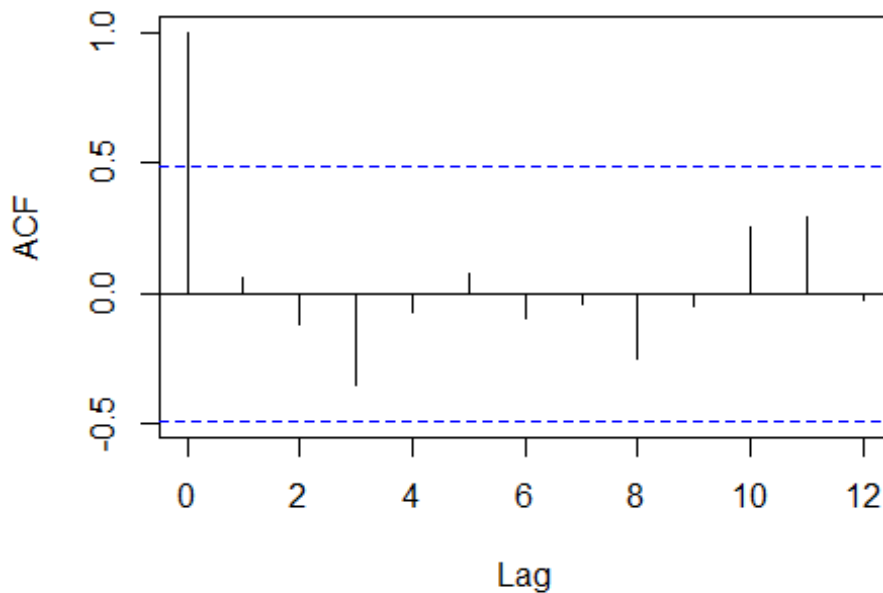
## Residuos Modelo Cuadratico



```
acf(residuos_cuadr, main = "ACF de los residuos")
```



### ACF de los residuos



```
cme = mean(residuos_cuadr ^ 2)
epam = mean(abs(residuos_cuadr / serie_desestacionalizada)) * 100
cat("CME:", cme, "\n")

## CME: 0.02584767

cat("EPAM:", epam, "\n")

## EPAM: 1.949349
```

### Concluye sobre el mejor modelo

Se elige como el mejor modelo el modelo cuadrático, puesto que este es el que mejor se ajusta al comportamiento de los datos desestacionalizados. De igual manera, se elige este modelo ya que se desempeñó mejor en las pruebas de CME y EPAM. El modelo cuadrático tuvo valores menores en estas pruebas, lo que indica un mejor rendimiento.

### Realiza el pronóstico para el siguiente año y gráficalo junto con los pronósticos previos y los datos originales.

```
library(forecast)

## Warning: package 'forecast' was built under R version 4.4.2

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

```

modelo_arima = auto.arima(ventas)

proximos_meses = data.frame(time = (length(ventas) + 1):(length(ventas) +
4))
predicciones = predict(modelo_cuadr, proximos_meses)
print(predicciones)

##          1          2          3          4
## 7.295777 7.331009 7.353779 7.364089

plot(time, serie_desestacionalizada, col = "blue", type = "o", pch = 19,
lwd = 2,
      xlab = "Tiempo", ylab = "Ventas", main = "Modelo Cuadrático de
Ventas", xlim=c(1, 20))

# Agregar la línea del modelo cuadrático en rojo
lines(time, predict(modelo_cuadr), col = "red", lwd = 2)

# Graficar las predicciones futuras en verde
lines((length(ventas) + 1):(length(ventas) + 4), predicciones, col =
"purple", lwd = 2, type = "o", pch = 19)

# Agregar Leyenda
legend("topleft", legend = c("Ventas", "Modelo Cuadrático", "Predicciones
Futuras"),
      col = c("blue", "red", "purple"), lwd = 2, pch = 19)

```

## Modelo Cuadrático de Ventas

