8-series-de-tiempo

November 16, 2023

1 8. Series de tiempo no estacionarias

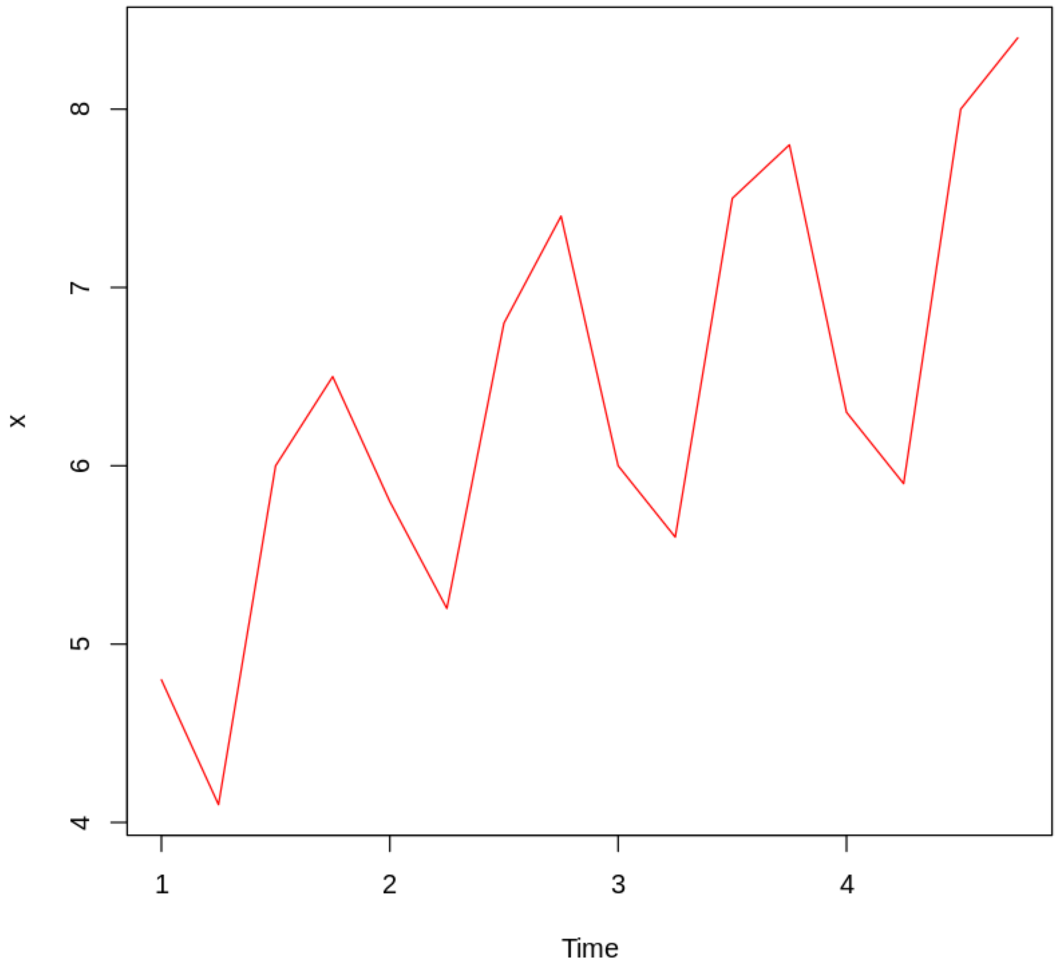
Francisco Mestizo Hernández A01731549

1.1 Ventas de televisores

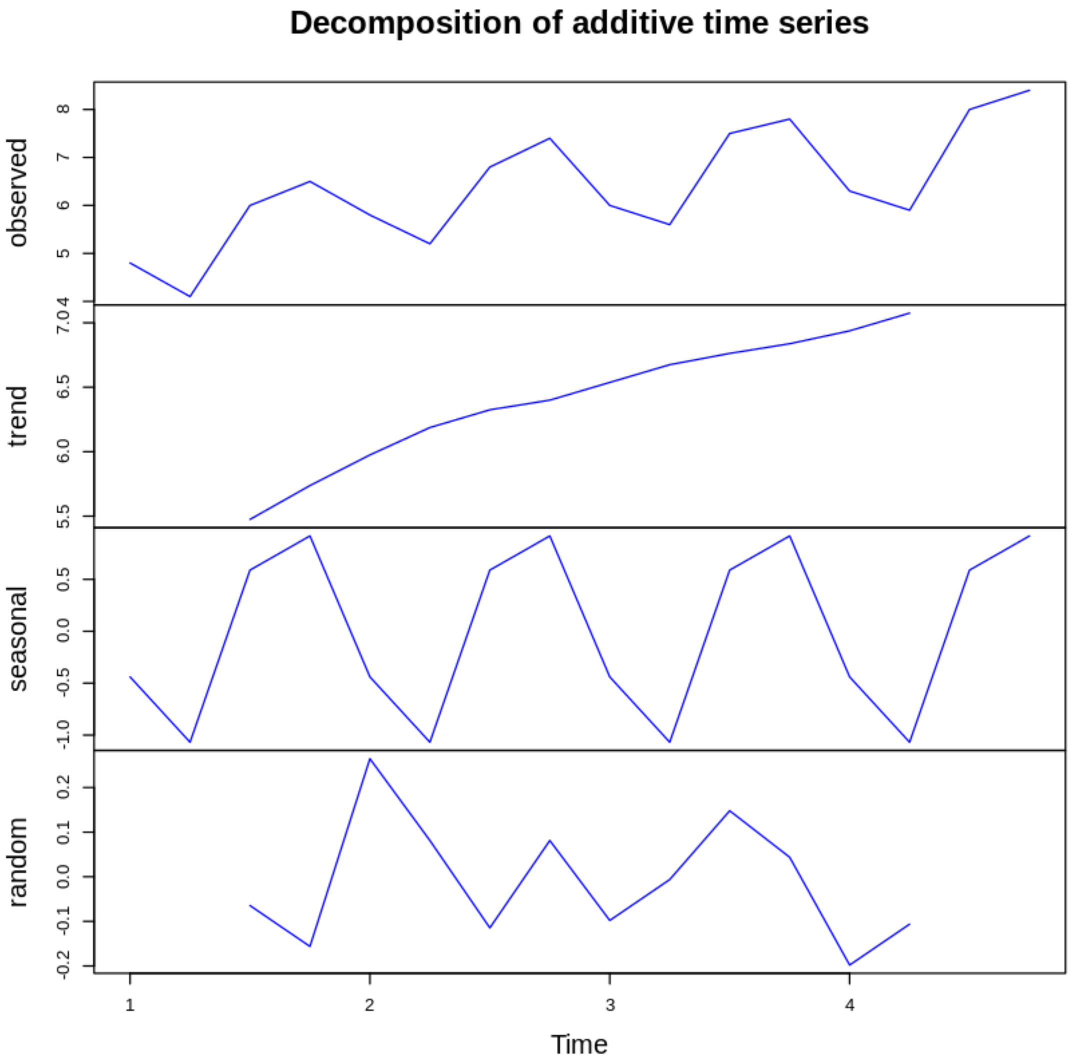
1.1.1 *Aditivo*

Primero, hacemos un análisis de los datos, graficandolos. Podemos ver que hay tres ciclos en la gráfica. Y también podemos ver que tiene una tendencia lineal ascendente.

[2]:
```
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)))
plot.ts(x, col = "red")
T = decompose(x)
plot(T, col ="blue")
```



x
4
5
6
7
8
1
2
3
4
Time



Y aqui podemos la estacionalidad de los datos

[3]:
```
T$seasonal
```

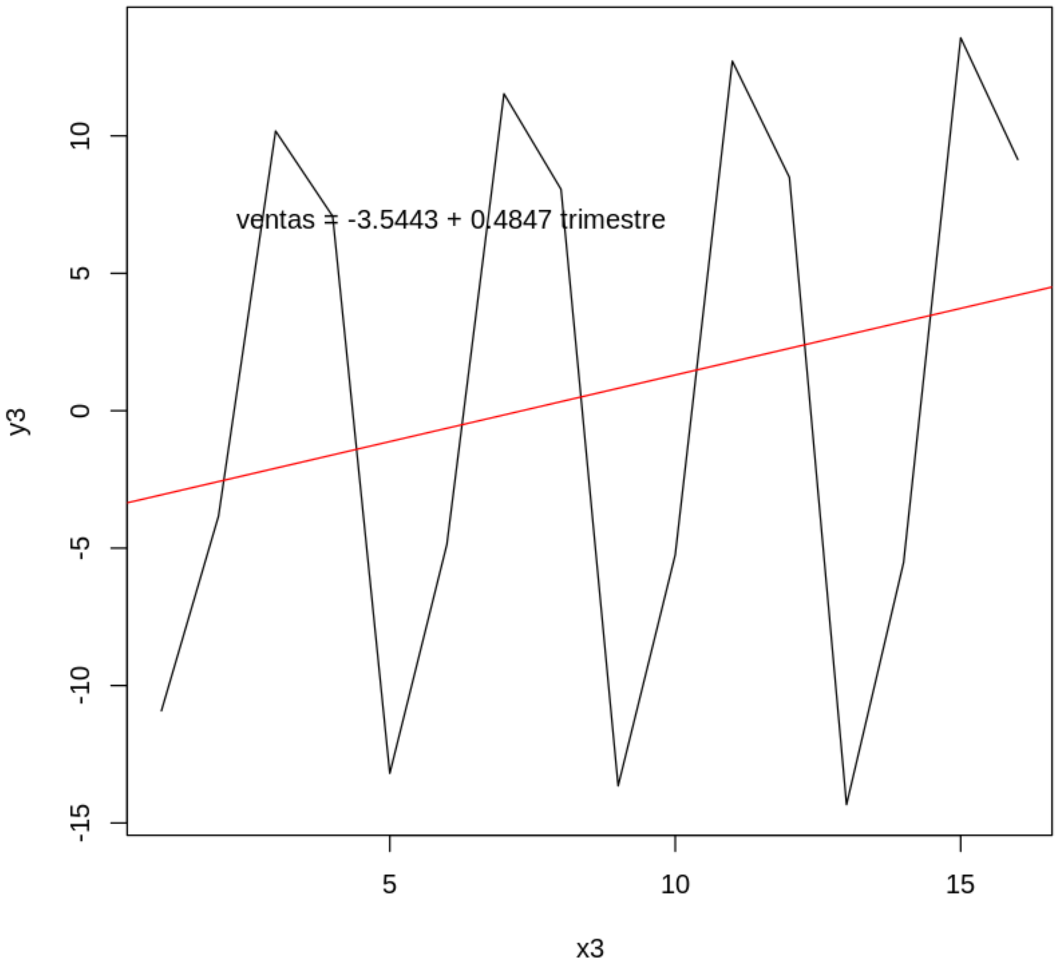
A Time Series: 4×4

	Qtr1	Qtr2	Qtr3	Qtr4
1	-0.4395833	-1.0687500	0.5895833	0.9187500
2	-0.4395833	-1.0687500	0.5895833	0.9187500
3	-0.4395833	-1.0687500	0.5895833	0.9187500
4	-0.4395833	-1.0687500	0.5895833	0.9187500

Podemos generar un modelo lineal para ver como se comporta con las ventas desestacionalizadas (después del aplanamiento). El modelo se puede ver gráficado abajo.

[4]:
```r
ventas_desestacionalizadas = (T$x)/(T$seasonal)
x3 = 1:16
y3 = ventas_desestacionalizadas
N3 = lm(y3~x3)
N3
plot(x3, y3, type = "l")
abline(N3, col = "red")
text(6, 7, " ventas = -3.5443 + 0.4847 trimestre")
```

```
Call:
lm(formula = y3 ~ x3)

Coefficients:
(Intercept)           x3
    -3.5443       0.4847
```



Aún asi, cuando obtenemos los detalles del modelo vemos que las variables no son significativas. Además, que la R cuadrada es muy pequeña

[5]:
```
summary(N3)
```

```
Call:
lm(formula = y3 ~ x3)

Residuals:
    Min      1Q  Median      3Q     Max
-17.088  -8.085   1.836   8.971  12.267

Coefficients:
```

```
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -3.5443     5.5166  -0.642    0.531
x3            0.4847     0.5705   0.850    0.410

Residual standard error: 10.52 on 14 degrees of freedom
Multiple R-squared:  0.04902,        Adjusted R-squared:  -0.0189
F-statistic: 0.7217 on 1 and 14 DF,  p-value: 0.4099
```

Las predicciones que obtenemos con este modelo lineal serían las siguientes.

[6]:
```
f = function(x) {-3.5443 + 0.4847*x}
# Los ídices estacionales son:
a1 = T$seasonal[1]
a2 =T$seasonal[2]
a3 = T$seasonal[3]
a4 = T$seasonal[4];
f(17)*a1*1000
f(18)*a2*1000
f(19)*a3*1000
f(20)*a4*100
```

-2064.1075

-5536.445625

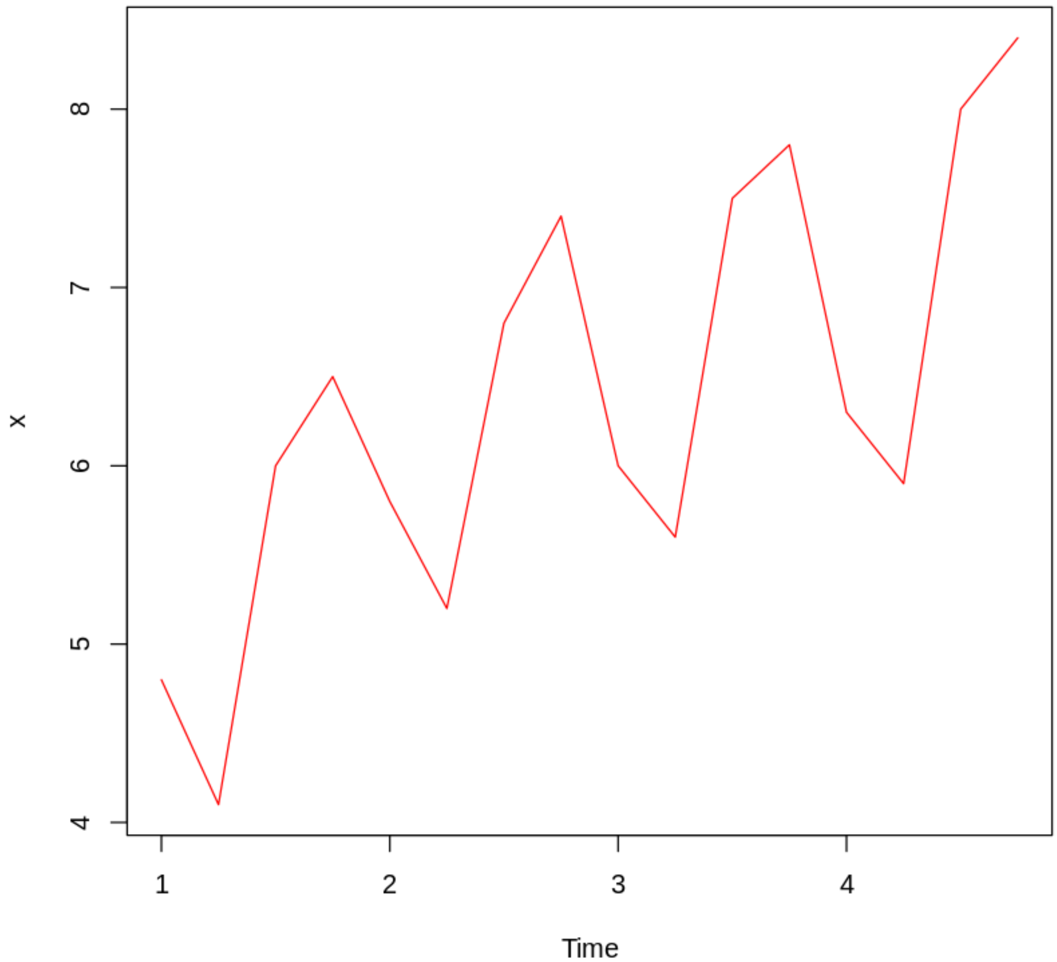
3339.98958333334

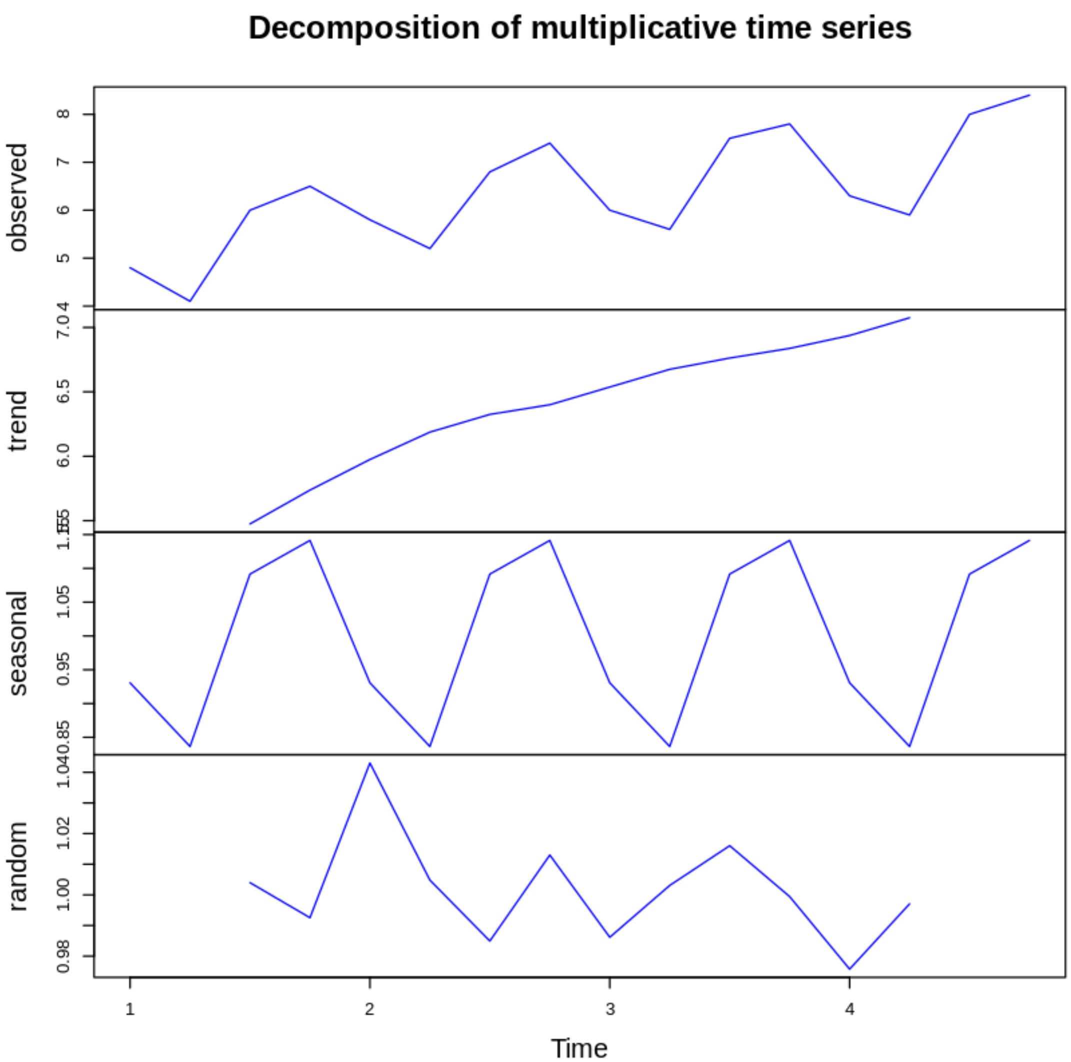
565.0036875

1.1.2 *Multiplicativo*

Podemos intentar hacer el modelo multiplicativo en lugar de aditivo. Veamos los resultados

[7]:
```
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)))
plot.ts(x, col = "red")
T = decompose(x, type="m")
plot(T, col ="blue")
```


x
4
5
6
7
8
1
2
3
4
Time

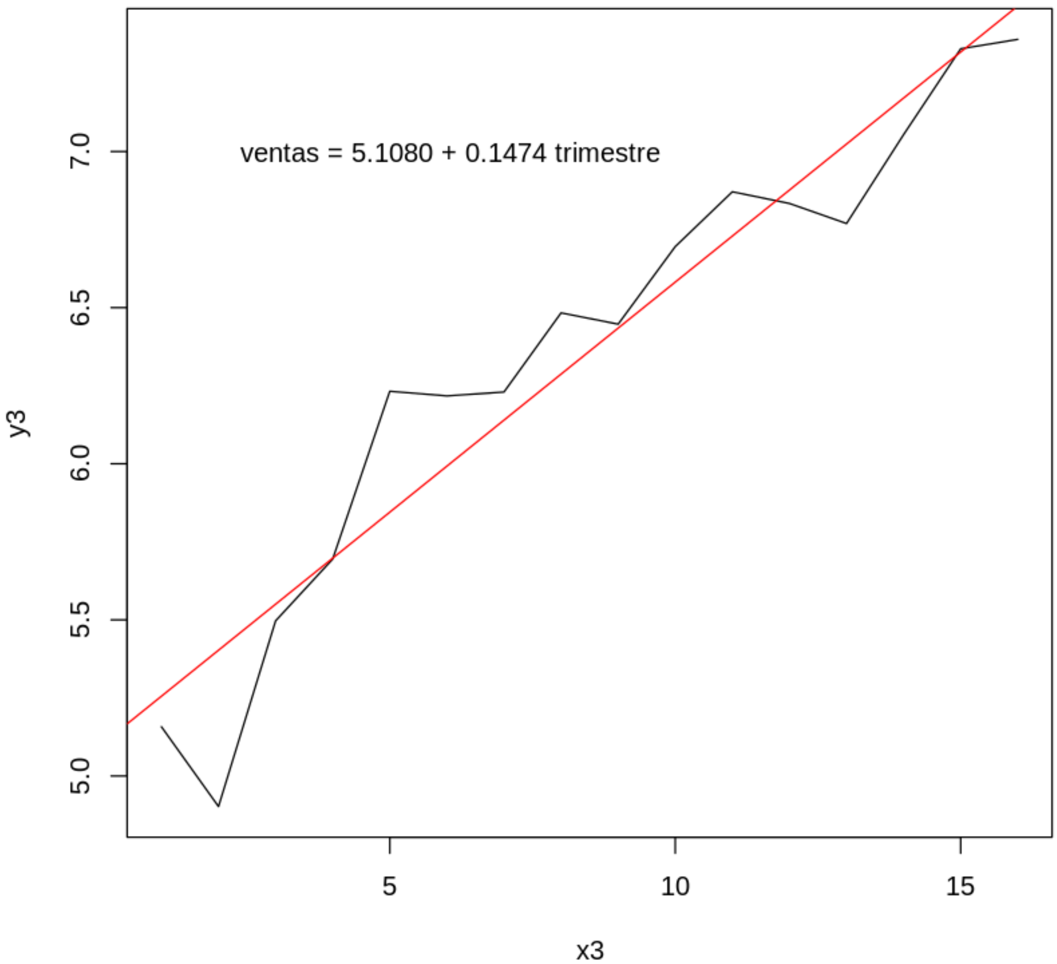


En la gráfica podemos ver que la linea se ajusta mucho mejor a los datos, por lo tanto es muy probable que este modelo sea mejor.

[8]:
```
ventas_desestacionalizadas = (T$x)/(T$seasonal)
x3 = 1:16
y3 = ventas_desestacionalizadas
N3 = lm(y3~x3)
N3
plot(x3, y3, type = "l")
abline(N3, col = "red")
text(6, 7, " ventas = 5.1080 + 0.1474 trimestre")
```

```
Call:
```

```
lm(formula = y3 ~ x3)

Coefficients:
(Intercept)           x3
     5.1080       0.1474
```



Si vemos los detalles del modelo comprobamos que las variables son significativas y que la R cuadrada aumento muchísimo, hasta tener un valor de 0.92

[9]:
```
summary(N3)
```

```
Call:
```

```
lm(formula = y3 ~ x3)

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 ***
x3           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
```

Y las predicciones que se obtienen son las siguientes. Podemos ver que tienen más sentido que las obtenidas en el primer modelo. Las predicciones que se muestran son para el año siguiente (los siguientes cuatro trimestres)

[10]:
```r
f = function(x) {5.1080 + 0.1474*x}
# Los ídices estacionales son:
a1 = T$seasonal[1]
a2 = T$seasonal[2]
a3 = T$seasonal[3]
a4 = T$seasonal[4]
pred1 = f(17)*a1*1000
pred2 = f(18)*a2*1000
pred3 = f(19)*a3*1000
pred4 = f(20)*a4*1000
pred1
pred2
pred3
pred4
```

7085.8720956619

6491.28393942345

8632.58539314895

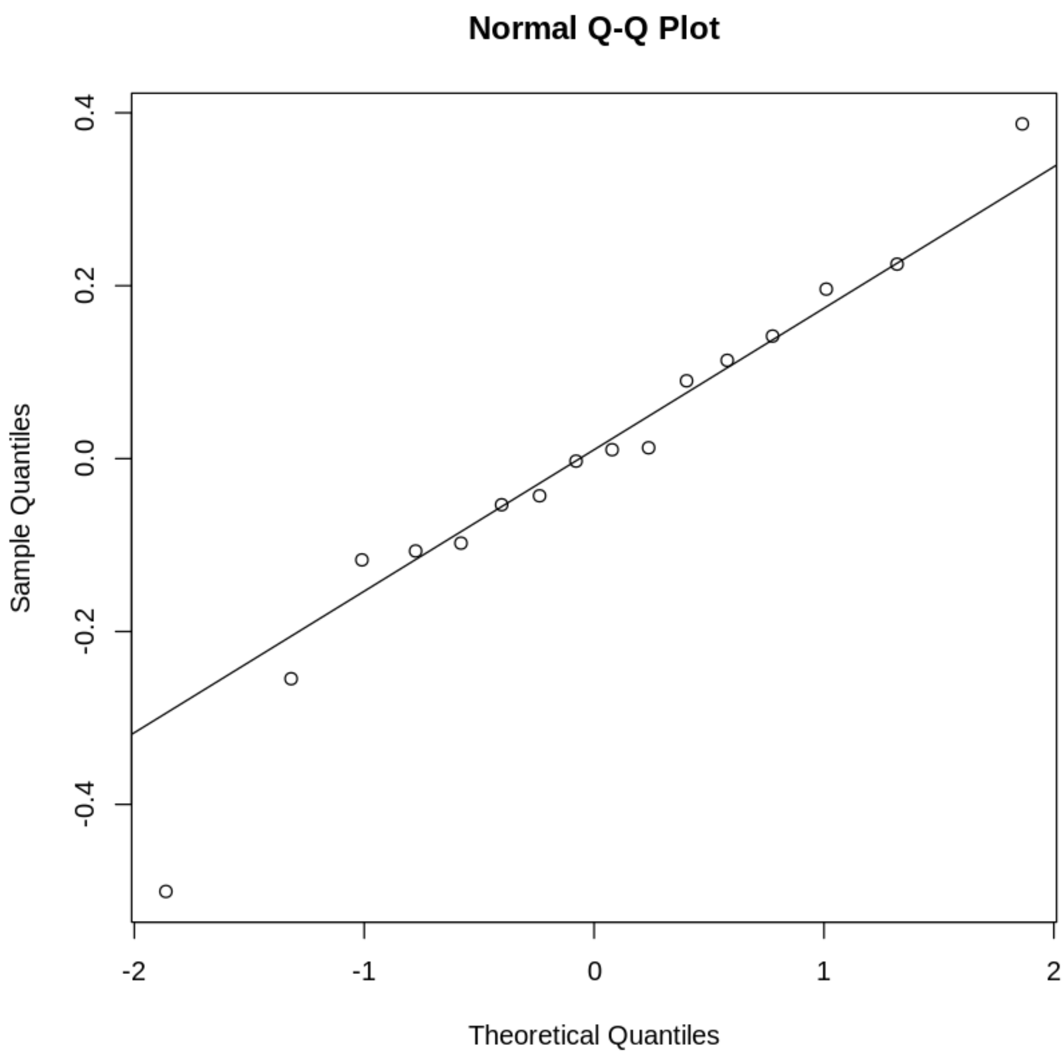
9195.26262943307

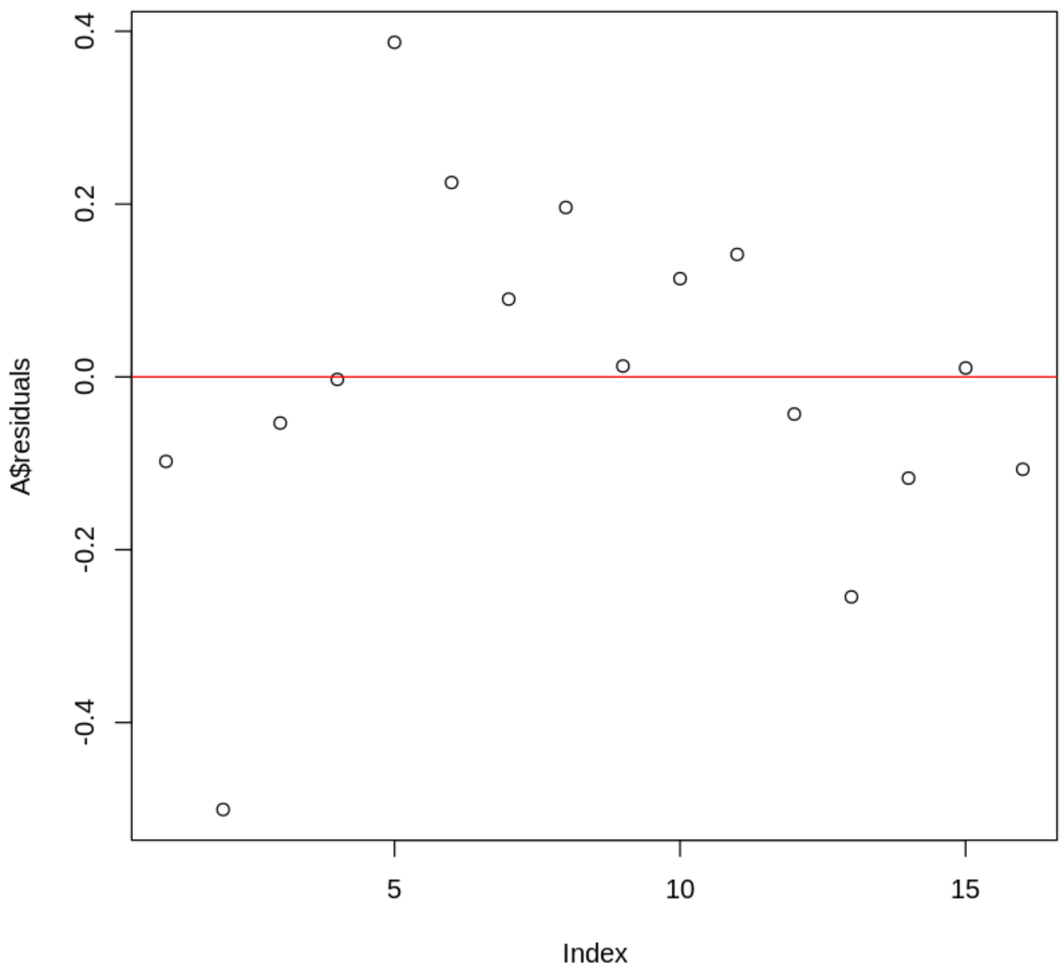
1.1.3 *Validez del modelo*

Podemos ver si el modelo que obtuvimos el valido se le harán pruebas de validez.

[11]:
```r
A = N3
shapiro.test(A$residuals)

#Gráficas auxiliares:
qqnorm(A$residuals)
qqline(A$residuals)

plot(A$residuals)
abline(h = 0, col = "red")
```

```
	Shapiro-Wilk normality test

data:  A$residuals
W = 0.96379, p-value = 0.7307
```


Normal Q-Q Plot
Sample Quantiles
0.4
0.2
0.0
-0.2
-0.4
-2
-1
0
1
2
Theoretical Quantiles



El resultado que obtenemos del test de shapiro indica normalidad. Se puede ver que en el gráfico de la Q-Q Plot los datos estan sobre las lineas pero las colas sí quedan un poco separadas. Además, los residuos nos muestran claramente un comportamiento cuadrático, ya que al inicio y al final están por debajo de la linea, pero en medio estan por encima. Esto puede indicar que es mejor que utilicemos otro modelo, como el cuadrático. Esto lo probaremos más adelante.

1.1.4 *CME2 y EPAM*

También se pueden comprobar los errores del modelo, tanto el cuadratico como el porcentual.

[12]:
```
predicciones = NA
e = NA
EPAM_todos = NA
for(i in 1:16){
```

```
  predicciones[i] = f(i)*T$seasonal[i]
  e[i] = ventas[i]-predicciones[i]
  EPAM_todos[i] =  abs(e[i]) / ventas[i]
}
```

[13]:
```
CME2=mean(ventas_desestacionalizadas^2,na.rm="TRUE")
EPAM = mean(EPAM_todos)
CME2
EPAM
```

40.960922120052

0.0243939555957905

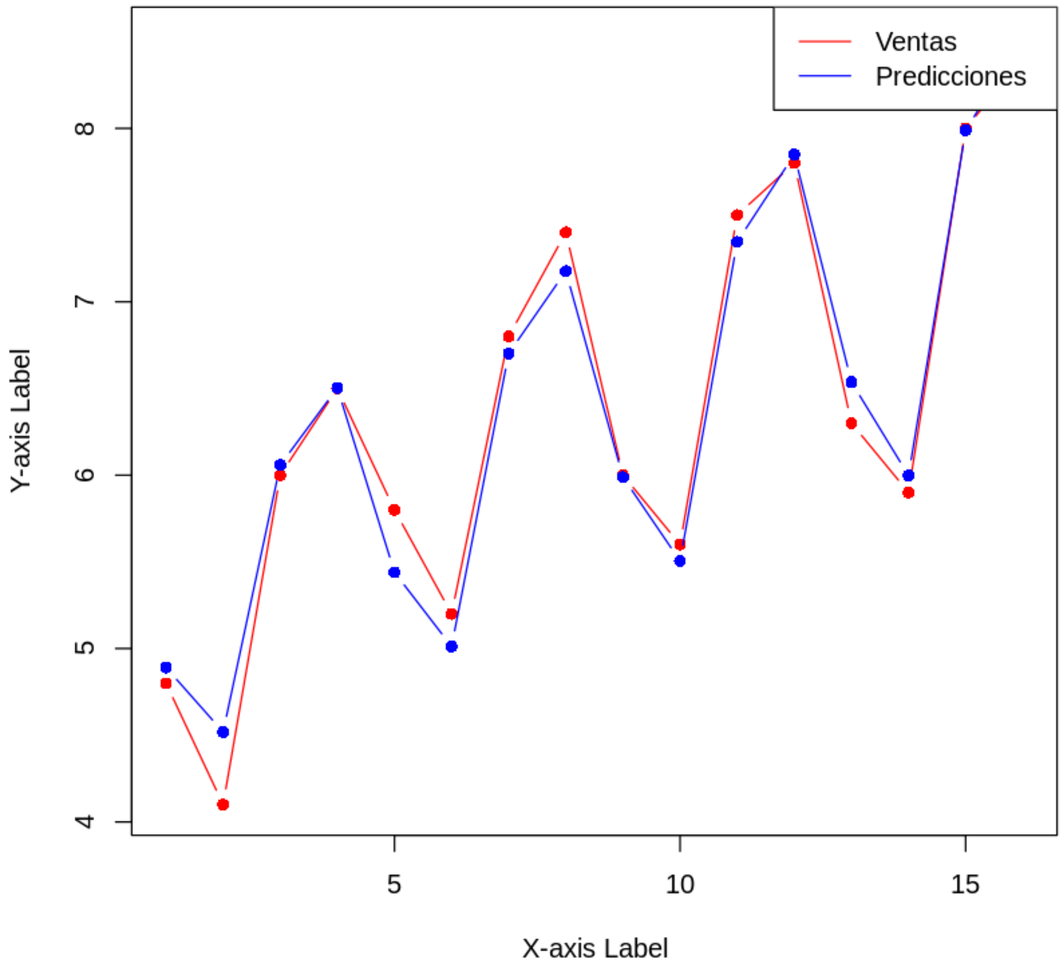
Los resultados que obtenemos es que el modelo explica el 98% de los casos ya que solo tiene un error del 2%. Esto nos puede indicar que es un modelo bastante bueno.

1.1.5 *Predicciones*

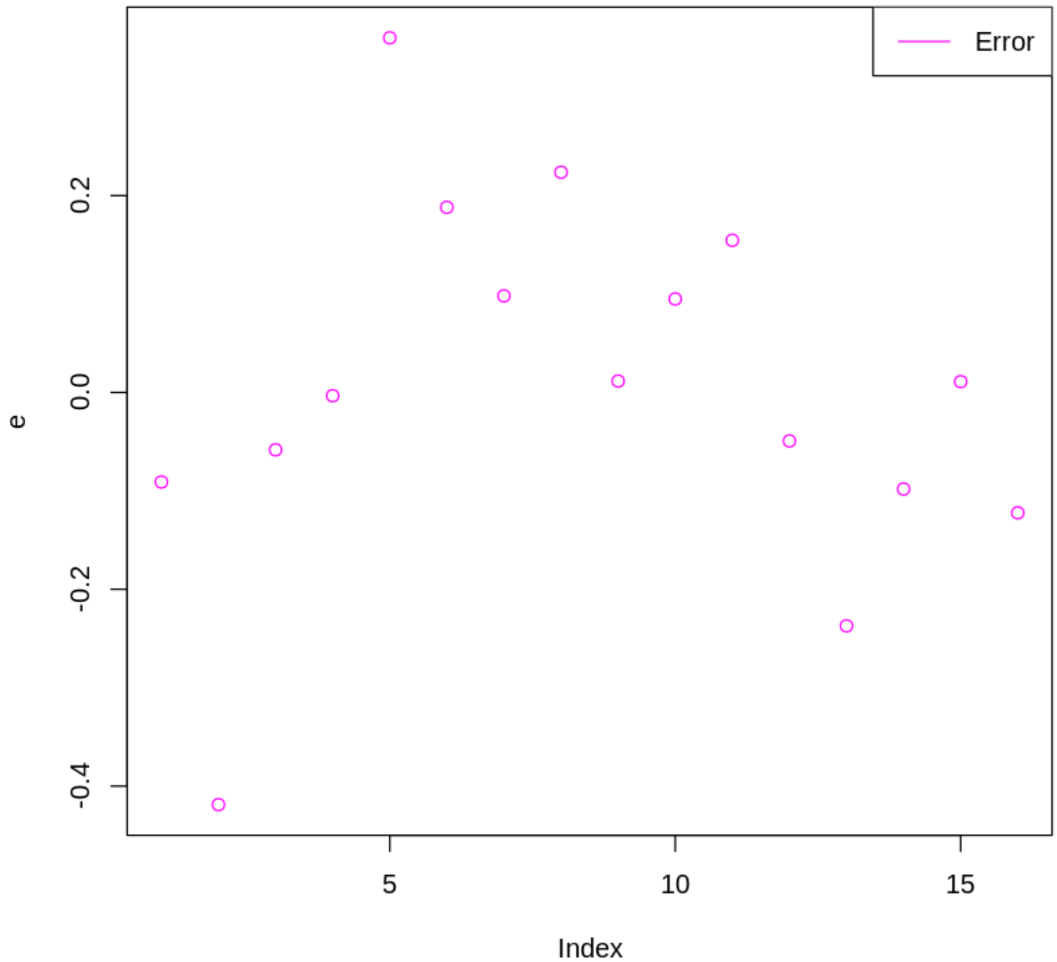
Con las predicciones que se realizaron para el segundo modelo (multiplicativo) podemos ver que tan cerca quedan de los valores reales. En general, las predicciones son bastante cercanas a la realidad.

Además, podemos volver a ver el error, donde confirmamos que tiene un comportamiento cuadratico, como habiamos visto arriba.

[14]:
```
plot(1:16, ventas, col = "red", type = "b", pch = 16, lty = 1, ylim =␣
 ↪c(min(c(ventas, predicciones)), max(c(ventas, predicciones))), xlab =␣
 ↪"X-axis Label", ylab = "Y-axis Label")
lines(1:16, predicciones, col = "blue", type = "b", pch = 16, lty = 1)
legend("topright", legend = c("Ventas", "Predicciones"), col = c("red",␣
 ↪"blue"), lty = 1)

plot(e, col = "magenta")
legend("topright", legend = c("Error"), col=c("magenta"), lty=1)
```



Ventas
Predicciones
Y-axis Label
4
5
6
7
8
5
10
15
X-axis Label



1.1.6 *Un mejor modelo: Cuadrático*

Como vimos arriba, aunque el modelo era bueno, parecia que se podría hacer mucho mejor, podemos probar hacer el mismo procedimiento pero ahora con un modelo cuadratico.

[15]:
```
x= ts(ventas, frequency = 4, start(c(2016,1)))
T = decompose(x, type="m")
```

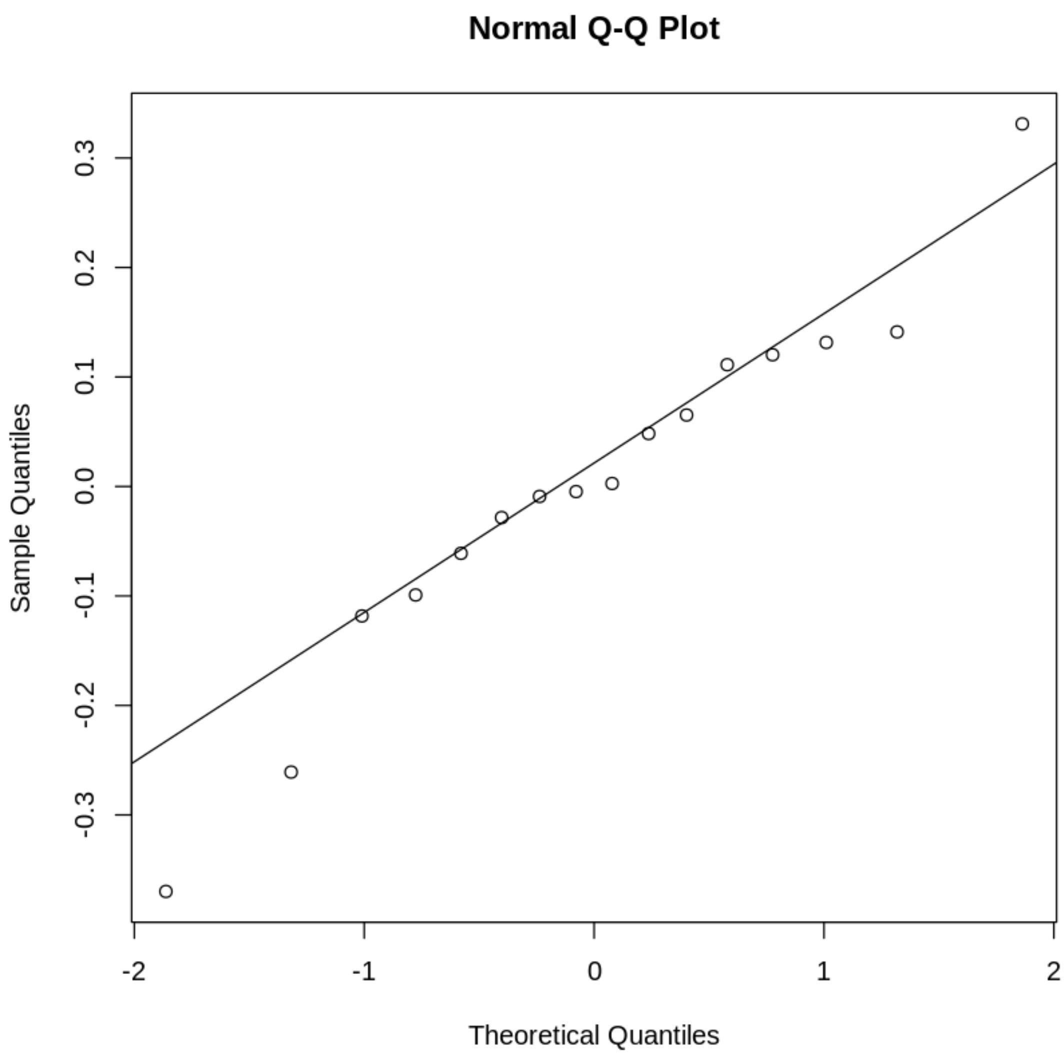
[16]:
```
ventas_desestacionalizadas = (T$x)/(T$seasonal)
x3 = 1:16
y3 = ventas_desestacionalizadas
N3 = lm(y3 ~ x3 + I(x3^2))
summary(N3)
```

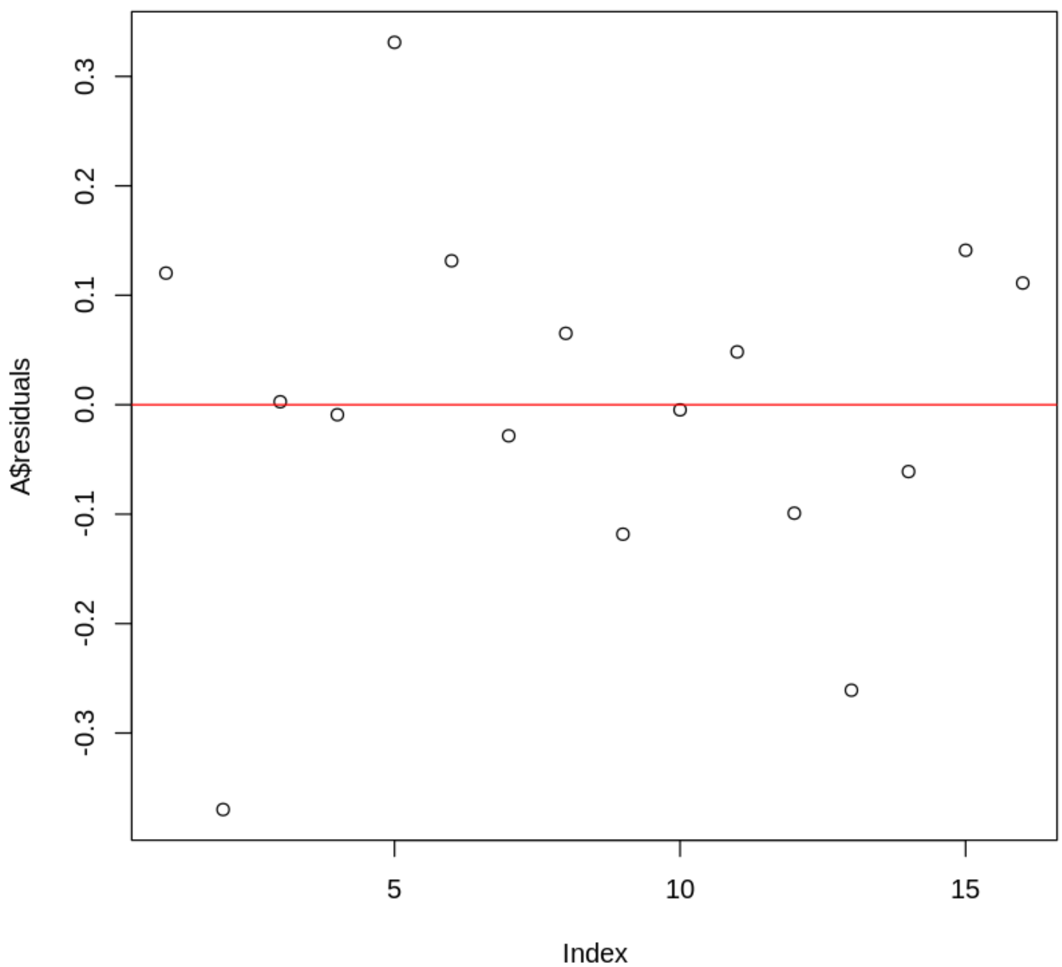
```
Call:
lm(formula = y3 ~ x3 + I(x3^2))

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 ***
x3           0.253302   0.041269   6.138 3.56e-05 ***
I(x3^2)     -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
```

Podemos ver que todas las variables son significativas y obtuvimos una R cuadrada todavia mejor. Subió de 0.915 a 0.94

[17]:
```r
A = N3
shapiro.test(A$residuals)

#Gráficas auxiliares:
qqnorm(A$residuals)
qqline(A$residuals)

plot(A$residuals)
abline(h = 0, col = "red")
```

```
	Shapiro-Wilk normality test

data:  A$residuals
W = 0.96004, p-value = 0.6625
```


Normal Q-Q Plot
Sample Quantiles
Theoretical Quantiles
0.3
0.2
0.1
0.0
-0.1
-0.2
-0.3
-2
-1
0
1
2



Además las colas ya no están tan alejadas y los residuos ya no parecen presentar un comportamiento. Aún así, quizás este modelo todavía podría ser mejorado. Y el test de shapiro nos indica normalidad

[18]:
```
f = function(x) {4.790283 + 0.253302*x -0.006231*x*x}
# Los ídices estacionales son:
a1 = T$seasonal[1]
a2 = T$seasonal[2]
a3 = T$seasonal[3]
a4 = T$seasonal[4]
pred1 = f(17)*a1*1000
pred2 = f(18)*a2*1000
pred3 = f(19)*a3*1000
pred4 = f(20)*a4*1000
```

```
pred1
pred2
pred3
pred4
```

6789.78951925353

6131.37029704438

8026.81118334748

8405.31355113241

Finalmente, con este nuevo modelo podemos hacer una prediccion para el siguiente año.

1.2 Un problemilla más: Libros de texto

1) Encuentre los promedios móviles de cuatro trimestres y los promedios móviles centrados

[19]:
```
anio1 <- c(1690, 940, 2625, 2500)
anio2 <- c(1800, 900, 2900, 2360)
anio3 <- c(1850, 1100, 2930, 2615)

# Suma móvil de cuatro trimestres
promediosMoviles <- (anio1 + anio2 + anio3) / 3
# Promedio móvil centrado de cuatro trimestres
promediosMovilesCentrados <- filter(c(anio1, anio2, anio3), rep(1, 3))/3

cat("Promedio Móvil de Cuatro Trimestres:", promediosMoviles, '\n\n')

cat("Promedio Móvil Centrado de Cuatro Trimestres:", promediosMovilesCentrados)
```

```
Promedio Móvil de Cuatro Trimestres: 1780 980 2818.333 2491.667

Promedio Móvil Centrado de Cuatro Trimestres: NA 1751.667 2021.667 2308.333
1733.333 1866.667 2053.333 2370 1770 1960 2215 NA
```

2) Calcule los índices estacionales de los cuatro trimestres

[20]:
```
# Promedios móviles centrados de cuatro trimestres
promedio_movil_centrado_cuatro_trimestres <- filter(ventas, rep(1, 3))/3

# Índices estacionales
indices_estacionales <- ventas / promedio_movil_centrado_cuatro_trimestres

# Imprimir resultados
print("Índices Estacionales:")
print(indices_estacionales)
```

```
[1] "Índices Estacionales:"
Time Series:
```

```
Start = 1
End = 16
Frequency = 1
 [1]        NA 0.8255034 1.0843373 1.0655738 0.9942857 0.8764045 1.0515464
 [8] 1.0990099 0.9473684 0.8795812 1.0765550 1.0833333 0.9450000 0.8762376
[15] 1.0762332        NA
```

3) ¿Cuándo obtiene la editorial el mayor índice estacional? ¿Parece razonable este resultado? ¿Por qué?

[21]:

```
# Encontrar el trimestre con el mayor índice estacional
trimestre_max_estacional <- which.max(indices_estacionales)

# Imprimir resultados
print("Trimestre con el Mayor Índice Estacional:")
print(trimestre_max_estacional)

# Obtener el año y trimestre correspondiente
anio_correspondiente <- ceiling(trimestre_max_estacional / 4)
trimestre_correspondiente <- trimestre_max_estacional %% 4
if (trimestre_correspondiente == 0) trimestre_correspondiente <- 4

print("Año y Trimestre Correspondiente:")
print(paste("Año", anio_correspondiente, ", Trimestre", 
 ↪trimestre_correspondiente))
```

```
[1] "Trimestre con el Mayor Índice Estacional:"
[1] 8
[1] "Año y Trimestre Correspondiente:"
[1] "Año 2 , Trimestre 4"
```

El resultado que obtenemos es que el trimestre con mayor inice estacional es el tercero del año. Esto tiene sentido, ya que es el trimestre en el que inician los ciclos escolares de la mayoría de las escuelas. Por lo tanto, hay más compras de libros de texto.