

0.0.0.0.1. Modelo de Regresión de Poisson

Dado que la variable respuesta es discreta y de tipo conteo, se ha elegido este modelo en el que se asume que el volumen de ventas diario sigue una distribución de Poisson.

$$Y \sim Po(\mu), \quad \mu = \text{Número medio de ventas diario}$$

Partición de los datos:

Se ha tomado una partición de 80 % 20 % para datos de entrenamiento y testeo, con el objetivo de entrenar el modelo para posteriormente estudiar su rendimiento.

0.0.0.0.1.1. Modelado

En primer lugar, estimaremos los parámetros del modelo de regresión de Poisson, utilizando un conjunto de datos con variables dummy para el día de la semana y el mes del año, con la intención de representar la pertenencia de cada instancia a los distintos grupos.

También entrenaremos un modelo haciendo uso del conjunto de datos con las variables día de la semana y mes en forma de factor, para comprobar que modelo nos da unas mejores métricas.

Las variables explicativas son las siguientes:

- Variables dummy/factorizadas del día de la semana y el mes del año
- Precio medio con impuestos y descuento
- Día de la semana
- Mes del año

Ventas totales

```
ModeloP_TOT_dummy =
  glm(VENTAS~PRECIO_MEDIO_IMPUESTOS*DESCUENTO_MEDIO+
      (LUNES+MARTES+MIERCOLES+JUEVES+VIERNES+SABADO+DOMINGO)+
      (AGOSTO+SEPTIEMBRE+OCTUBRE+NOVIEMBRE+DICIEMBRE+ENERO),
      family = poisson(link = "log"),
      data=Ventas_TOTAL_ENT_poiss)
summary(ModeloP_TOT_dummy)
```

```
##
```

```
## Call:
```

```
## glm(formula = VENTAS ~ PRECIO_MEDIO_IMPUESTOS * DESCUENTO_MEDIO +
```

```
##      (LUNES + MARTES + MIERCOLES + JUEVES + VIERNES + SABADO +
```

```
##      DOMINGO) + (AGOSTO + SEPTIEMBRE + OCTUBRE + NOVIEMBRE +
```

```
##      DICIEMBRE + ENERO), family = poisson(link = "log"), data = Ventas_TOTAL_ENT_po
```

```
##
```

```
## Deviance Residuals:
```

```
##      Min        1Q    Median        3Q        Max
```

```
## -51.810   -6.571    1.080    4.994   64.707
```

```
##
```

```
## Coefficients: (2 not defined because of singularities)
```

```
##              Estimate Std. Error z value Pr(>|z|)
```

```
## (Intercept)      -32.219952    0.955000 -33.738  < 2e-16
```

```

## PRECIO_MEDIO_IMPUESTOS      26.098720    0.663672   39.325 < 2e-16
## DESCUENTO_MEDIO             2.655409    3.725073    0.713  0.476
## LUNES1                      2.503693    0.015988  156.601 < 2e-16
## MARTES1                     2.415363    0.016009  150.873 < 2e-16
## MIERCOLES1                  2.365094    0.016268  145.386 < 2e-16
## JUEVES1                     2.595014    0.016010  162.082 < 2e-16
## VIERNES1                    2.402489    0.016007  150.085 < 2e-16
## SABADO1                     2.500752    0.016010  156.204 < 2e-16
## DOMINGO1                    NA           NA         NA      NA
## AGOSTO1                     -0.254510  0.006886  -36.962 < 2e-16
## SEPTIEMBRE1                 -0.125003  0.006765  -18.479 < 2e-16
## OCTUBRE1                    0.108339    0.006135   17.660 < 2e-16
## NOVIEMBRE1                 -0.077361    0.007198  -10.748 < 2e-16
## DICIEMBRE1                 -0.034890    0.006418   -5.436 5.44e-08
## ENERO1                      NA           NA         NA      NA
## PRECIO_MEDIO_IMPUESTOS:DESCUENTO_MEDIO -1.825982  2.588410  -0.705  0.481
##
## (Intercept)                ***
## PRECIO_MEDIO_IMPUESTOS      ***
## DESCUENTO_MEDIO
## LUNES1                      ***
## MARTES1                     ***
## MIERCOLES1                  ***
## JUEVES1                     ***
## VIERNES1                    ***
## SABADO1                     ***
## DOMINGO1
## AGOSTO1                     ***
## SEPTIEMBRE1                 ***
## OCTUBRE1                    ***
## NOVIEMBRE1                 ***
## DICIEMBRE1                 ***
## ENERO1
## PRECIO_MEDIO_IMPUESTOS:DESCUENTO_MEDIO
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##      Null deviance: 102206  on 144  degrees of freedom
## Residual deviance:  25019  on 130  degrees of freedom
## AIC: 26371
##
## Number of Fisher Scoring iterations: 4

```

En la salida, podemos ver en la columna *estimate* la estimación de los coeficientes de regresión para las distintas variables, indicando, para las variables numéricas, el cambio medio en el número de ventas que se produciría si aumentáramos en una unidad esa variable, y para las variables discretas, el cambio medio que provocaría en el número de

ventas el que la variable tomara o no ese valor. También encontramos una columna para el error estándar, el valor del estadístico Z y el p-valor.

Todas las variables que se refieren a meses o días de la semana influyen en el volumen de ventas:

- Con respecto al día de la semana, sabemos que el día de la semana que más ventas hay es el Sábado, por lo que estudiamos como afectan las ventas si es o no este día del fin de semana. $\hat{\beta}_{\text{Sábado}} = 2.427548 > 0$, es decir, el volumen de ventas aumentará, en media $e^{2.427548} = 11$ unidades si la compra se realiza un Sábado
- El mes donde hubo más ventas fué durante el mes de Octubre, y según este modelo el volumen de ventas aumentará, en media, manteniendo el resto de variables constante en $e^{2.138172} = 8$ unidades si la compra se hace durante este mes.

Además, la interacción entre las variables descuento y precio medio también es significativa, es decir, la asociación que existe entre ambas varía en función de los diferentes valores que tomen.

Por otro lado, no podemos considerar que algunos días del mes como los días 10, 13, 14 o 15 sean significativos en el modelo de cara a predecir el volumen de ventas diario para un nivel de significación del 95 %.

Vamos ahora a entrenar el modelo para las variables factorizadas:

```
ModeloP_TOT_factor= glm(VENTAS~PRECIO_MEDIO_IMPUESTOS+
                        DIA_SEMANA*MES+DESCUENTO_MEDIO,
                        family = poisson(link = "log"),
                        data=Ventas_TOTAL_ENT)
summary(ModeloP_TOT_factor)
```

```
##
```

```
## Call:
```

```
## glm(formula = VENTAS ~ PRECIO_MEDIO_IMPUESTOS + DIA_SEMANA *
```

```
## MES + DESCUENTO_MEDIO, family = poisson(link = "log"), data = Ventas_TOTAL_ENT
```

```
##
```

```
## Deviance Residuals:
```

```
##      Min        1Q      Median        3Q        Max
```

```
## -44.212   -3.094    0.479    3.585   58.763
```

```
##
```

```
## Coefficients:
```

```
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -29.399611    1.113717 -26.398 < 2e-16 ***
## PRECIO_MEDIO_IMPUESTOS 25.846617    0.772717  33.449 < 2e-16 ***
## DIA_SEMANA2     -0.040739    0.015208  -2.679 0.007391 **
## DIA_SEMANA3     -0.269271    0.019286 -13.962 < 2e-16 ***
## DIA_SEMANA4      0.039848    0.021983   1.813 0.069883 .
## DIA_SEMANA5      0.060860    0.013909   4.376 1.21e-05 ***
## DIA_SEMANA6      0.086613    0.013002   6.662 2.71e-11 ***
## DIA_SEMANA7     -2.802053    0.048116 -58.235 < 2e-16 ***
## MES8            -0.004214    0.014360  -0.293 0.769186
## MES9             0.076174    0.015402   4.946 7.58e-07 ***
## MES10           -0.089625    0.015596  -5.747 9.10e-09 ***
```

```

## MES11          -0.073488    0.013666   -5.378  7.55e-08 ***
## MES12          -0.003946    0.015205   -0.260  0.795238
## DESCUENTO_MEDIO    0.063439    0.008285    7.657  1.90e-14 ***
## DIA_SEMANA2:MES8   -0.179717    0.022960   -7.827  4.99e-15 ***
## DIA_SEMANA3:MES8   -0.090149    0.025918   -3.478  0.000505 ***
## DIA_SEMANA4:MES8   -0.415099    0.028027  -14.811 < 2e-16 ***
## DIA_SEMANA5:MES8   -0.309014    0.021824  -14.159 < 2e-16 ***
## DIA_SEMANA6:MES8   -0.435314    0.020480  -21.256 < 2e-16 ***
## DIA_SEMANA7:MES8    0.329373    0.061599    5.347  8.94e-08 ***
## DIA_SEMANA2:MES9   -0.221695    0.022066  -10.047 < 2e-16 ***
## DIA_SEMANA3:MES9    0.113470    0.024429    4.645  3.40e-06 ***
## DIA_SEMANA4:MES9   -0.338998    0.028216  -12.015 < 2e-16 ***
## DIA_SEMANA5:MES9   -0.598435    0.023369  -25.608 < 2e-16 ***
## DIA_SEMANA6:MES9   -0.195576    0.022563   -8.668 < 2e-16 ***
## DIA_SEMANA7:MES9    0.719879    0.062623   11.495 < 2e-16 ***
## DIA_SEMANA2:MES10    0.193201    0.022172    8.714 < 2e-16 ***
## DIA_SEMANA3:MES10    0.281635    0.025615   10.995 < 2e-16 ***
## DIA_SEMANA4:MES10    0.565819    0.026597   21.274 < 2e-16 ***
## DIA_SEMANA5:MES10    0.021127    0.020549    1.028  0.303888
## DIA_SEMANA6:MES10    0.177425    0.020087    8.833 < 2e-16 ***
## DIA_SEMANA7:MES10    0.506864    0.060592    8.365 < 2e-16 ***
## DIA_SEMANA2:MES11   -0.111932    0.021915   -5.107  3.27e-07 ***
## DIA_SEMANA3:MES11   -0.033636    0.033058   -1.018  0.308910
## DIA_SEMANA4:MES11   -0.062049    0.026811   -2.314  0.020651 *
## DIA_SEMANA5:MES11   -0.024749    0.019503   -1.269  0.204439
## DIA_SEMANA6:MES11    0.140983    0.019330    7.293  3.02e-13 ***
## DIA_SEMANA7:MES11    0.023367    0.061702    0.379  0.704904
## DIA_SEMANA2:MES12    0.005424    0.021705    0.250  0.802669
## DIA_SEMANA3:MES12    0.201484    0.024466    8.235 < 2e-16 ***
## DIA_SEMANA4:MES12    0.149292    0.026123    5.715  1.10e-08 ***
## DIA_SEMANA5:MES12   -0.187272    0.021809   -8.587 < 2e-16 ***
## DIA_SEMANA6:MES12   -0.501937    0.024909  -20.151 < 2e-16 ***
## DIA_SEMANA7:MES12    0.303646    0.060343    5.032  4.85e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##      Null deviance: 102206  on 144  degrees of freedom
## Residual deviance:  19286  on 101  degrees of freedom
## AIC: 20696
##
## Number of Fisher Scoring iterations: 5

```

- La variable *descuento* puede considerarse significativa, y según la estimación del coeficiente $\hat{\beta}_{\text{descuento}} = 0.02494 > 0$ podemos afirmar lo siguiente: manteniendo el resto de variables constantes, un aumento de la variable descuento en una unidad, hará que el volumen de ventas aumente en: $e^{0.02494} = 1$ unidades.

DATOS	R2	RMSE
Datos entrenamiento	0.7066326	593.1505
Datos Test	0.2395861	777.0907

- En este caso, para la variable Sábado podemos decir que si el día de la semana es sábado y manteniendo constantes el resto de variables, el volumen de ventas disminuirá en $e^{-0.17516} = 1$ unidad.
- Para el mes de Octubre, y según este modelo, el volumen de ventas aumentará, en media, manteniendo el resto de variables constante en $e^{11.95380} = 1.55407 \times 10^5$ unidades si la compra se hace durante este mes.
- Además, la interacción entre las variables día de la semana y mes del año es significativa, indicando que la asociación que existe entre ambas varía en función de los distintos valores que toman.

Para seleccionar un modelo adecuado, se ha hecho uso del criterio de información de Akaike, eligiendo aquel que nos de el menor AIC. A continuación se exponen los correspondientes valores del AIC para ambos modelos:

- Modelo con variables dummy: AIC=26371
- Modelo con variables factorizadas: AIC=20696

El modelo seleccionado es el modelo para variables factorizadas, que nos da el menor valor del AIC, indicación de una mayor calidad del modelo estadístico.

A continuación, procedemos a entrenar el modelo seleccionado en los datos de entrenamiento haciendo uso de la función *predict*

A continuación mostramos una tabla con las métricas que nos dan el rendimiento del modelo

El modelo obtenido explica el 70.66 % de la variabilidad total de los datos en el conjunto de entrenamiento, y el 23.96 % para los datos de testeo, no pudiendo considerar este como un buen modelo para explicar el volumen de ventas total. El valor del error cuadrático medio indica que el modelo se va a equivocar de media, en 777 ventas, que es un valor alto para el volumen de ventas que se está considerando, ya que hay días que se venden menos de 700 items.

Ventas de productos con calcio

```
ModeloP_CALCIO_dummy =
  glm(VENTAS~PRECIO_MEDIO_IMPUESTOS+
      (LUNES+MARTES+MIERCOLES+JUEVES+VIERNES+SABADO+DOMINGO)+
      (AGOSTO+SEPTIEMBRE+OCTUBRE+NOVIEMBRE+DICIEMBRE+ENERO),
      family = poisson(link = "log"),
      data=Ventas_CALCIO_ENT_poiss)
#summary(ModeloP_CALCIO_dummy)
```

```
ModeloP_CALCIO_factor=
  glm(VENTAS~PRECIO_MEDIO_IMPUESTOS+DIA_SEMANA*MES,
      family = poisson(link = "log"),
```

DATOS_C	R2_C	RMSE_C
Datos entrenamiento	0.6519377	359.9727
Datos Test	0.1703046	489.6647

```
data=Ventas_CALCIO_ENT)
#summary(ModeloP_CALCIO_factor)
```

Para ambos modelos podemos afirmar:

- La variable precio medio es altamente significativa, un aumento de esta variable en un euro indicará una leve disminución de las ventas
- Gran parte de las variables son significativas en el modelo

Además, para el modelo entrenado para el conjunto de datos con variables factorizadas, la interacción entre el día de la semana y el mes es significativa, indicando una variación de las ventas en función de las distintas combinaciones de valores de estas variables.

De nuevo, el modelo seleccionado es el modelo para variables factorizadas, que nos da el menor valor del AIC, indicación de una mayor calidad del modelo estadístico.

- $AIC_{dummies} = 17366$
- $AIC_{factorizacion} = 13618$

Entrenamos el modelo seleccionado en los datos de entrenamiento haciendo uso de la función *predict*

En el siguiente gráfico podemos ver una representación de los valores observados respecto de los valores ajustados.

Volvemos a tener unos resultados pobres, el modelo es capaz de explicar el 0.1703046 % de la variabilidad del volumen de ventas para los datos de testeo, a pesar de explicar el 0.6519377 % de la variabilidad para los datos de entrenamiento, es decir, el modelo ha “aprendido” los datos con los que ha entrenado pero no generaliza bien para nuevos datos.

Ventas de productos sin calcio

```
ModeloP_SIN_CALCIO_dummy =
  glm(VENTAS~PRECIO_MEDIO_IMPUESTOS+
      (LUNES+MARTES+MIERCOLES+JUEVES+VIERNES+SABADO+DOMINGO)+
      (AGOSTO+SEPTIEMBRE+OCTUBRE+NOVIEMBRE+DICIEMBRE+ENERO),
      family = poisson(link = "log"),
      data=Ventas_SIN_CALCIO_ENT_poiss)
#summary(ModeloP_SIN_CALCIO_dummy)
```

```
ModeloP_SIN_CALCIO_factor=
  glm(VENTAS~PRECIO_MEDIO_IMPUESTOS+DESCUENTO_MEDIO+
      DIA_SEMANA*MES,
      family = poisson(link = "log"),
      data=Ventas_SIN_CALCIO_ENT)
#summary(ModeloP_SIN_CALCIO_factor)
```

DATOS_SC	R2_SC	RMSE_SC
Datos entrenamiento	0.7117361	264.3933
Datos Test	0.2298473	385.9581

Para ambos modelos podemos afirmar:

- La variable precio medio es altamente significativa en ambos modelos, indicando que un aumento de esta variable en un euro indicará una leve disminución de las ventas
- Gran parte de las variables son significativas en el modelo

Además, para el modelo entrenado para el conjunto de datos con variables factorizadas, la interacción entre el día de la semana y el mes es significativa, indicando una variación de las ventas en función de las distintas combinaciones de valores de estas variables.

De nuevo, el modelo seleccionado es el modelo para variables factorizadas, que nos da el menor valor del AIC, indicación de una mayor calidad del modelo estadístico.

- $AIC_{dummies} = 12738$
- $AIC_{factorizacion} = 10153$

Entrenamos el modelo seleccionado en los datos de entrenamiento haciendo uso de la función *predict*.

Volvemos a tener unos resultados no demasiado buenos. Este modelo no es capaz de explicar más que el 0.2298473 % de la variabilidad del volumen de ventas en los datos de testeo, a pesar de explicar el 0.7117361 % para los datos de entrenamiento. Los errores cuadráticos medios son muy elevados en comparación con el volumen de ventas que se está prediciendo.

0.0.0.0.2. Análisis de Series Temporales

Se consideró aplicar un análisis de series temporales debido a la estructura de los datos, ya que este tipo de análisis contempla la estructura temporal de los mismos. Como ya se avanzó en el desarrollo teórico, aplicaremos la metodología Box-Jenkins, la cual tiene en cuenta la dependencia existente de los datos, construyendo así un modelo ARIMA.

Trataremos de modelizar el volumen de ventas total según día de la semana. Para construir la serie, primero hemos añadido los días 25 de Diciembre y 1 de Enero con un número de ventas 0, ya que, si no se tomaba esta decisión, la serie ya no estaría definida según la realidad.

0.0.0.0.2.1. Creación ST y representación de los datos

Los datos que se han utilizado para el análisis han sido las ventas totales para cada día, agrupando los datos según día de la semana, obteniendo así datos con período $S=7$.

Sin embargo, si construimos la serie con los valores actuales, no podremos aplicarle transformaciones, en particular la transformación de Box-Cox, ya que existen dos valores nulos, las ventas para los días 25 de Diciembre y 1 de Enero. Por este motivo, sumamos una constante a todas las observaciones de modo que sean todas positivas.

Nota: la constante que hemos sumado es de 10 unidades.

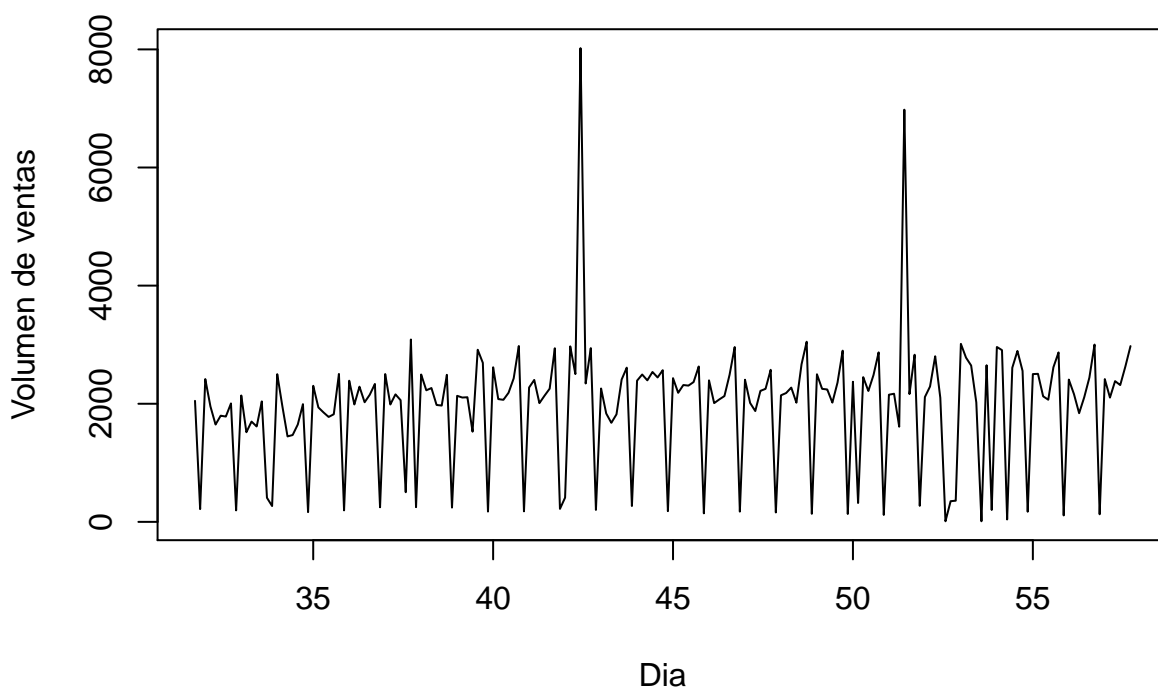
El primer dato, indica que el sábado de la semana 31 del año se vendieron un total de 2049 artículos, aunque en la realidad es que se vendieron 10 unidades menos, pero se le ha sumado una constante a la serie.

```
tsDiaSemanal = ts(Ventas_Totales_Dia_Semana_Completo$ArtVendidos,
                  frequency=7, # Período
                  start=c(31,6) # Semana 36, sábado
                  )
print(tsDiaSemanal,calendar=TRUE)
```

##	p1	p2	p3	p4	p5	p6	p7
## 31						2049	216
## 32	2418	1955	1648	1798	1783	2005	194
## 33	2140	1518	1698	1615	2040	408	268
## 34	2501	1957	1447	1470	1649	1993	165
## 35	2304	1938	1853	1776	1822	2505	194
## 36	2390	1988	2289	2025	2153	2335	247
## 37	2503	1987	2159	2060	502	3088	249
## 38	2495	2228	2266	1979	1969	2490	241
## 39	2134	2105	2110	1527	2914	2693	174
## 40	2618	2079	2066	2186	2429	2978	178
## 41	2275	2403	2010	2136	2255	2940	220
## 42	410	2972	2502	8021	2343	2942	202
## 43	2258	1835	1677	1821	2410	2611	268
## 44	2391	2495	2396	2538	2445	2568	181
## 45	2433	2185	2317	2307	2367	2631	144
## 46	2396	2010	2073	2130	2491	2960	173
## 47	2407	2013	1876	2218	2252	2575	159
## 48	2143	2181	2274	2018	2667	3050	136
## 49	2499	2252	2242	2019	2358	2899	133
## 50	2372	319	2450	2216	2486	2870	117
## 51	2153	2169	1611	6979	2164	2830	271
## 52	2113	2292	2804	2100	10	350	358
## 53	3014	2780	2647	2023	10	2652	201
## 54	2961	2908	40	2609	2894	2556	171
## 55	2502	2506	2126	2068	2617	2869	109
## 56	2409	2163	1840	2110	2443	3002	128
## 57	2417	2104	2383	2316	2631	2976	

Después de haber definido los datos como una serie temporal, visualizamos la evolución de la serie en el tiempo.

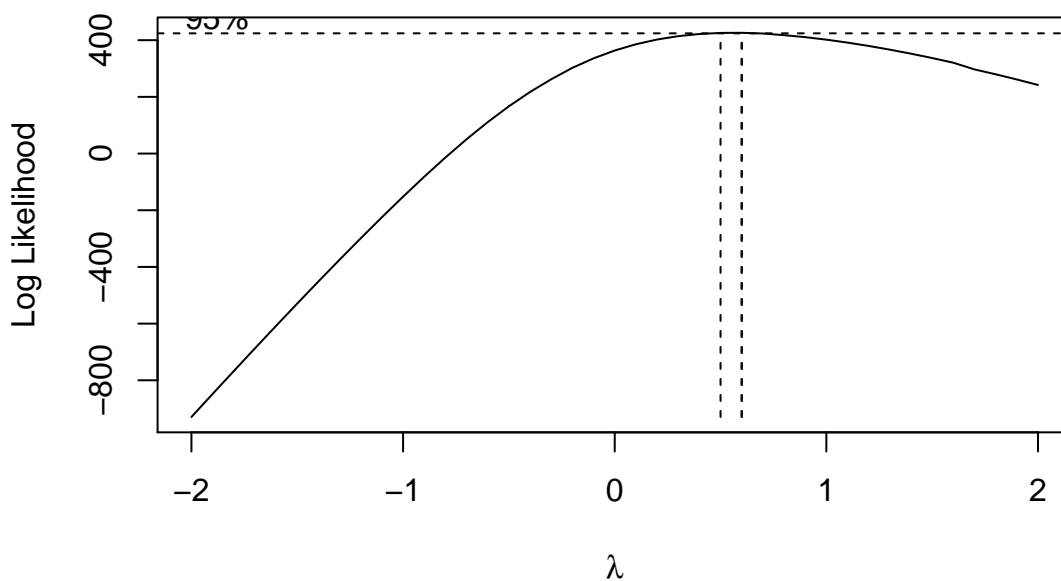
Volumen total de venta por según día de la semana



En el gráfico se puede apreciar cierta estacionalidad de los datos, es decir, movimientos que se repiten regularmente año tras año en los mismo períodos. También observamos que las oscilaciones van aumentando con el tiempo, indicando que la varianza no es constante. Por este motivo, debemos hacer alguna transformación para que la varianza sea constante en el tiempo.

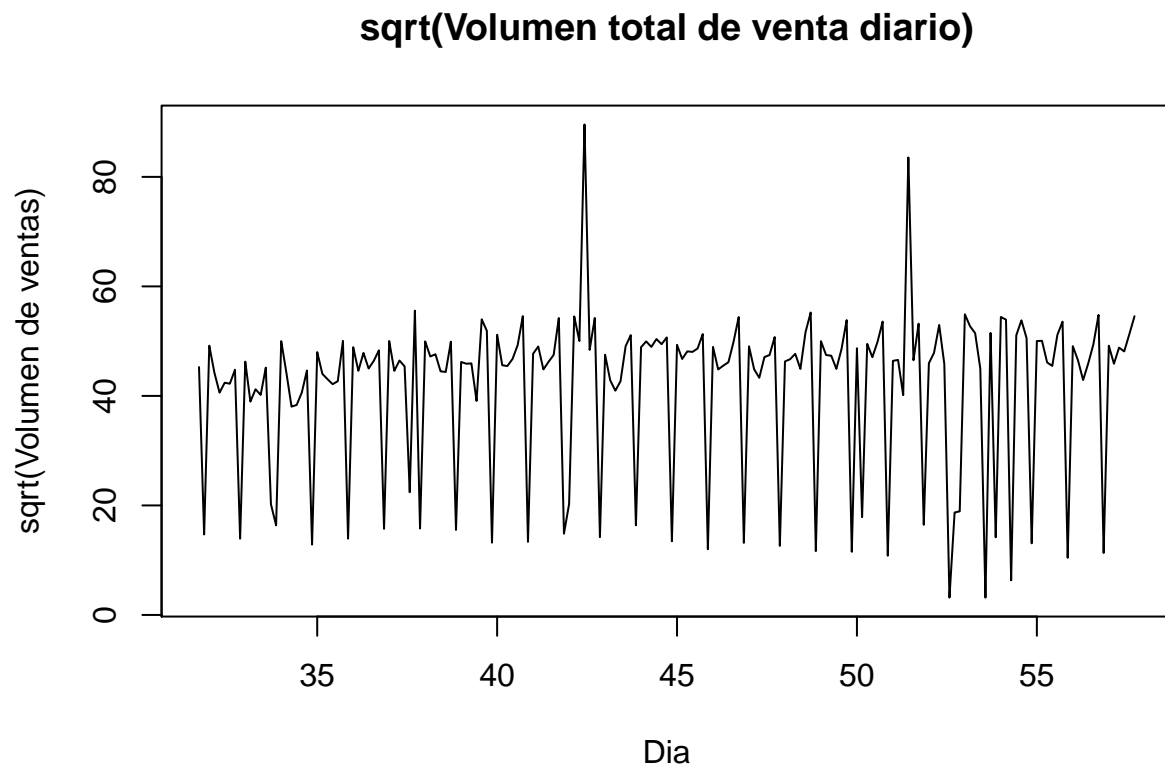
0.0.0.0.2.2. Transformación de BoxCox para estabilizar la varianza

Para encontrar una transformación que haga que la varianza sea constante en el tiempo, haremos uso de la familia de transformaciones Box-Cox con ayuda de la librería *TSA*.



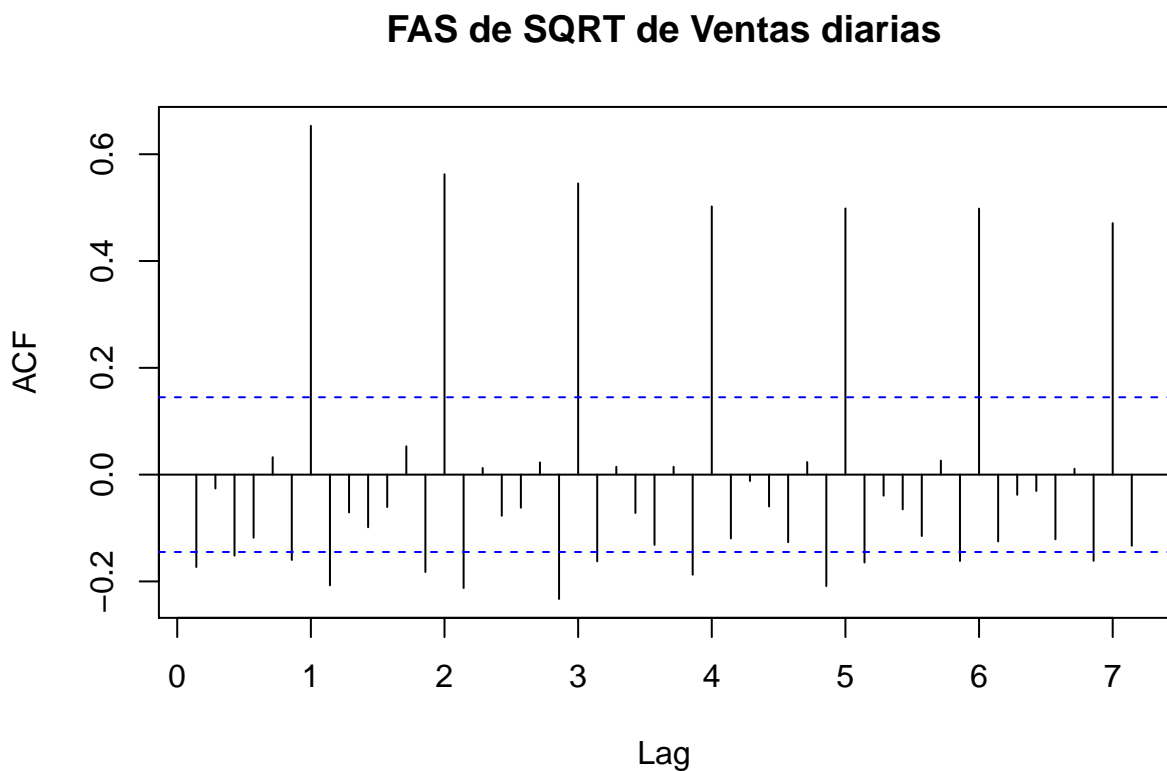
La función *BoxCox.ar* sugiere un óptimo de $\lambda = 0.6$, con un intervalo de confianza al 95 %: (0.5,0.6). Se necesita una transformación sencilla y comprensible, por lo que se ha obtenido por tomar como valor de lambda el extremo inferior del intervalo, $\lambda = 1/2$.

Transformamos los datos y volvemos a representar la serie.



0.0.0.0.2.3. Transformaciones para estabilizar la media

Vamos a estudiar si el motivo de la no estacionalidad de los datos en media se debe a que se trata de un proceso integrado. Para ello, hacemos uso de la función de autocorrelación simple.

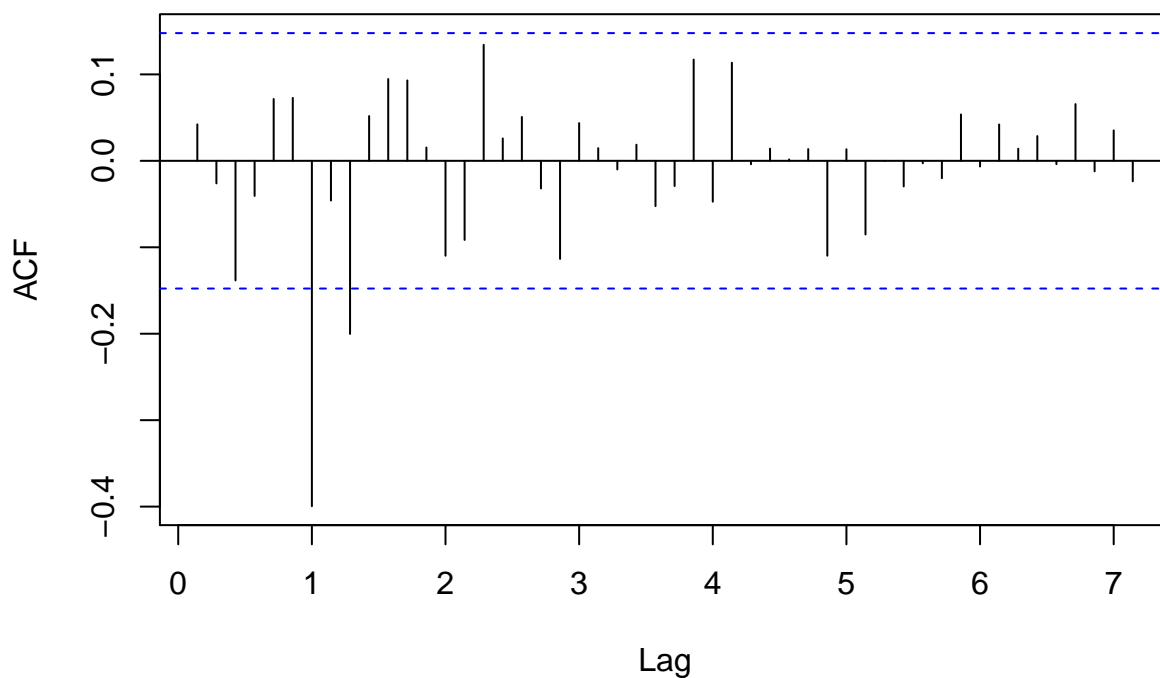


La FAS muestral decrece de lentamente en los retardos estacionales de período 7, indicando

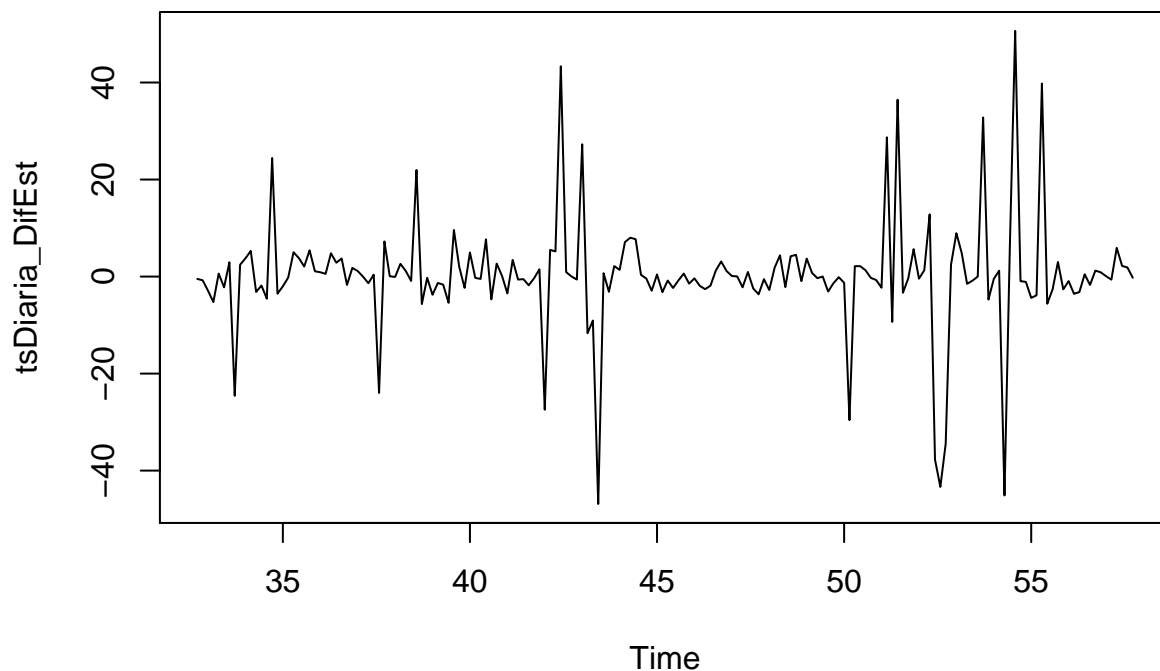
que estamos ante un modelo integrado. Debido a esta situación, hacemos una diferencia estacional de la serie y volvemos a representar la FAS ($s=7$).

```
tsDiaria_DifEst = diff(tsDiaSemanal_transf, lag=7, diff=1)
acf(tsDiaria_DifEst, main="FAS de primera diferencia estacional", lag=50)
```

FAS de primera diferencia estacional



Ahora la función de autocorrelación muestral corresponde a la de un proceso estacionario. Por último, representamos gráficamente la serie diferenciada:



Observamos que la serie no muestra ningún comportamiento en particular, sino que se aprecia aleatoriedad, por lo que se podría pensar, que nos encontramos ante un proceso estacionario. Ahora estamos en condiciones de buscar un modelo estacionario para la serie.

0.0.0.0.2.4. Contraste de estacionariedad

Para confirmar la estacionariedad de los datos sugerida con la observación de la gráfica, necesitamos aplicar un test de hipótesis. Aplicamos el test de raíz unitaria de Dikey-Fuller, donde se contrasta la estacionariedad de los datos a través del siguiente test de hipótesis:

$$\begin{cases} H_0 : \text{El polinomio autoregresivo tiene una raíz unitaria} \\ H_1 : \text{Todas las raíces del polinomio autoregresivo son estacionarias} \end{cases}$$

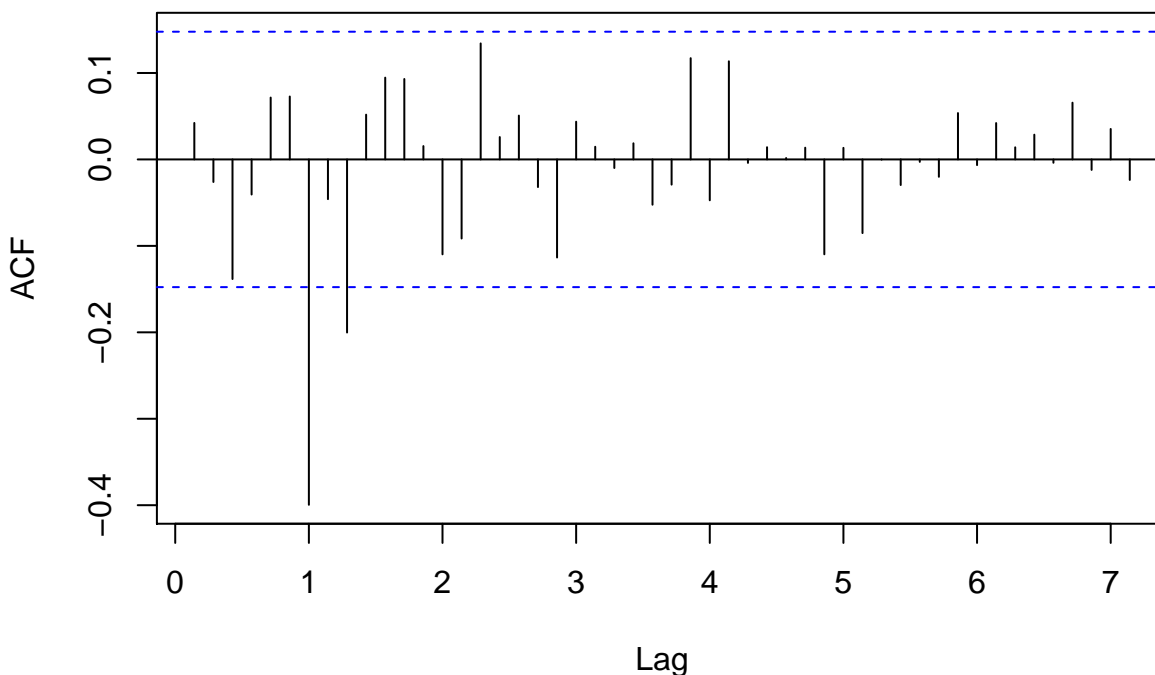
```
##
## Augmented Dickey-Fuller Test
##
## data: tsDiaria_DifEst
## Dickey-Fuller = -5.1008, Lag order = 5, p-value = 0.01
## alternative hypothesis: stationary
```

El p-valor del test= $0.01 < 0.05 = \alpha$, y por tanto concluimos que no existen evidencias significativas para asumir que el polinomio autoregresivo tiene alguna raíz unitaria, la serie es estacionaria.

0.0.0.0.2.5. Identificación de la estructura ARIMA de la serie

Trataremos de identificar la estructura ARIMA más adecuada para esta serie a través de la función de autocorrelación simple (FAC) y de la función de autocorrelación parcial (FAP). Determinar el modelo más adecuado consistirá en e identificar el orden de los procesos de medias móviles y autoregresivos de la componente estacional y la componente regular.

FAS tras una diferencia estacional

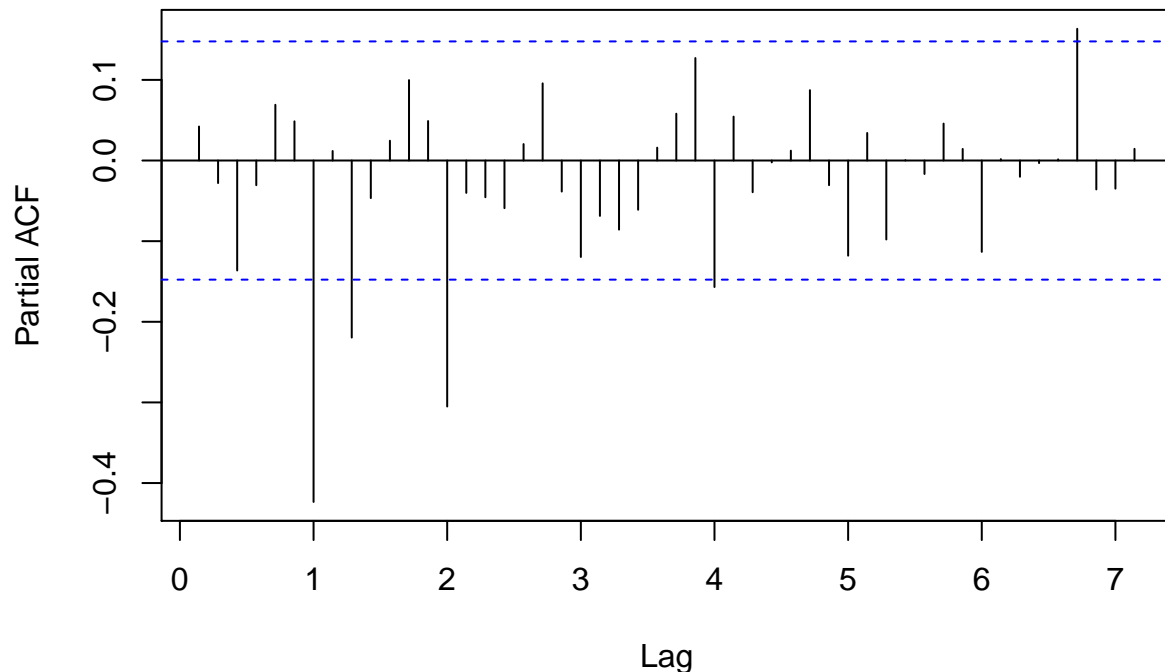


- Parte regular: En los primeros retardos no observamos ninguna autocorrelación significativamente no nula, indicando que el modelo tiene una estructura ARMA(0,0) en la parte regular.

- Parte estacional: Observamos una autocorrelación en el primer retardo estacional, por lo que parecería que la parte estacional tiene una estructura $MA(1)_{12}$.

Vamos a comprobar estos supuestos con la FAP.

FAP tras una diferencia estacional



- Parte regular: De nuevo, no hay autocorrelaciones significativamente no nulas en los primeros retardos.
- Parte estacional: En los retardos estacionales, observamos como las autocorrelaciones decrecen rápidamente y a su izquierda, no hay autocorrelaciones significativamente no nulas, lo que avalaría aún más la suposición de un $MA(1)$ en la parte estacional. Modelo propuesto: $MA(1)_{12}$

También observamos como hay otras autocorrelaciones significativamente no nulas, pero esto es debido a que se trata de un intervalo de confianza al 95 %, por lo que cabe esperar que haya algunas autocorrelaciones fuera de las bandas.

El modelo a considerar es un modelo estacional multiplicativo integrado de medias móviles puro: $ARIMA(0, 1, 1)_{12}$

0.0.0.0.2.6. Estimación de parámetros y diagnóstico del modelo

Una vez hemos obtenido un modelo, se han estimado sus parámetros con la función *arima*.

```
Ajuste1 = arima( tsDiaria_DifEst ,    # Serie tras una diferencia estacional
                seasonal = list(order=c(0,0,1),period=7 ))
```

```
Ajuste1
```

```
##
```

```
## Call:
```

```
## arima(x = tsDiaria_DifEst, seasonal = list(order = c(0, 0, 1), period = 7))
```

```
##
```

```
## Coefficients:
##          sma1  intercept
##          -1.0000    0.0898
## s.e.      0.0827    0.0871
##
## sigma^2 estimated as 79.04:  log likelihood = -645.71,  aic = 1295.42
```

Trás comprobar si los coeficientes estimados son o no significativamente nulos, procedemos a eliminar la media del modelo, obteniendo así uno donde todos los coeficientes son significativamente no nulos.

```
##              2.5 %    97.5 %
## sma1        -1.16213365 -0.8378657
## intercept -0.08097887  0.2605421
```

```
Ajuste1_1 = arima( tsDiaria_DifEst ,    # Serie tras una diferencia estacional
                  order = c(0,0,0),seasonal = list(order=c(0,0,1),period=7),
                  include.mean = FALSE # Eliminamos la media
                  )
Ajuste1_1
```

```
##
## Call:
## arima(x = tsDiaria_DifEst, order = c(0, 0, 0), seasonal = list(order = c(0,
##      0, 1), period = 7), include.mean = FALSE)
##
## Coefficients:
##          sma1
##          -0.9999
## s.e.      0.1009
##
## sigma^2 estimated as 79.52:  log likelihood = -646.24,  aic = 1294.47
confint(Ajuste1_1)
```

```
##              2.5 %    97.5 %
## sma1 -1.197648 -0.8021465
```

El modelo ajustado corresponde a la siguiente ecuación:

$$Y_t = (1 - L^7)(1 + 0.9999\Theta^7)\alpha_t, \quad \alpha_t \sim RB(0, \sigma^2)$$

Para comprobar si el modelo es o no adecuado, comprobamos su validez a través de la diagnosis de los residuos y concluimos que este ajuste no es adecuado, ya que según el Test de Ljung-Box, no existen evidencias significativas para aceptar la incorrelación de los residuos: $p\text{-valor} = 0.002524 < 0.05 = \alpha$. Además, gráficamente podemos observar que los residuos no se comportan como un ruido blanco.

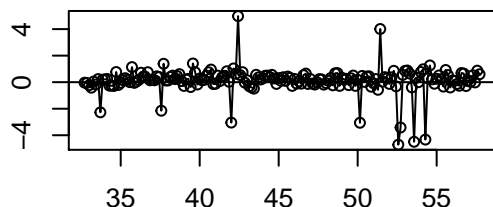
```
##
## Ljung-Box test
##
## data:  Residuals from ARIMA(0,0,0)(0,0,1)[7] with zero mean
```

```
## Q* = 31.855, df = 13, p-value = 0.002524
```

```
##
```

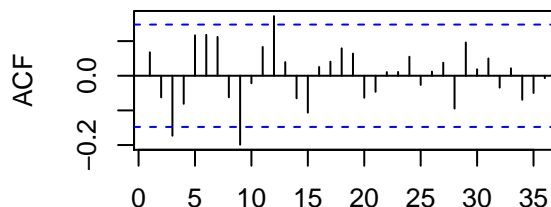
```
## Model df: 1. Total lags used: 14
```

Residuos estandarizados



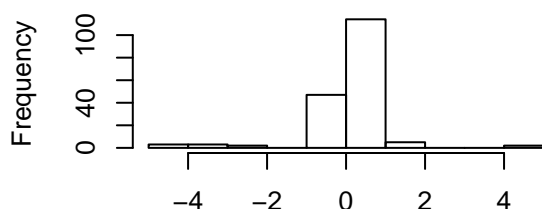
Time

FAS de los residuos



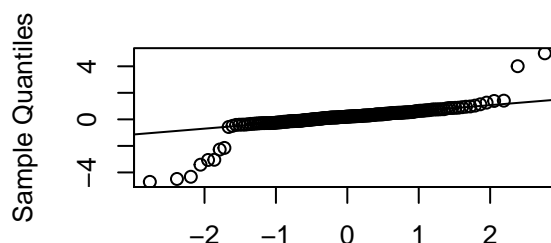
Lag

Histograma



Residuos estandarizados

Normal Q-Q Plot



Theoretical Quantiles

Vamos a probar otro modelo, en particular, a través del paquete *forecast* haciendo uso de la función *auto.arima*, que busca un modelo que minimiza el AIC.

```
## Series: tsDiaSemanal_transf
```

```
## ARIMA(1,0,0)(2,1,0)[7]
```

```
##
```

```
## Coefficients:
```

```
##          ar1          sar1          sar2
```

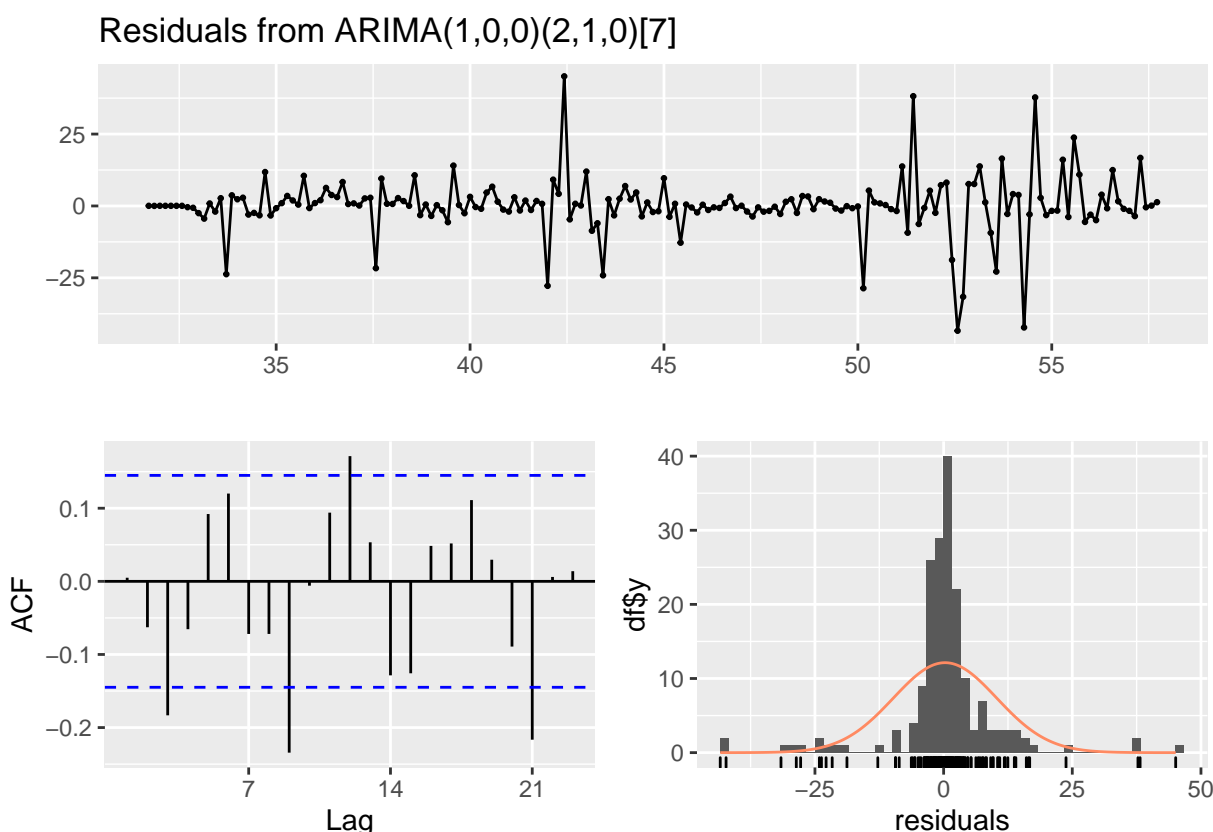
```
##          0.0709      -0.5180      -0.3068
```

```
## s.e.      0.0754      0.0708      0.0694
```

```
##
```

```
## sigma^2 = 106.2: log likelihood = -660.08
```

```
## AIC=1328.17 AICc=1328.4 BIC=1340.85
```



```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(1,0,0)(2,1,0)[7]
## Q* = 36.336, df = 11, p-value = 0.0001488
##
## Model df: 3.    Total lags used: 14
```

El ajuste propuesto es un modelo: $ARIMA(1, 0, 0) \times ARIMA(2, 0, 0)_7$, pero tampoco es adecuado, ya que volvemos a rechazar la hipótesis de incorrelación de los residuos del Test de Ljung-Box.

No hemos podido encontrar un modelo adecuado que se ajuste a los datos y que pase la diagnosis, ya que los residuos no provenían en ningún caso de un proceso de ruido blanco, es decir, no estaban incorrelados entre sí. Por este motivo, al no ser los retardos independientes, un retardo puede guardar cierta relación con otro retardo k períodos después. En estos casos, la autocorrelación puede conducir a una inexactitud en el modelo predictivo, que nos llevaría a interpretaciones erróneas.

La tabla mostrada a continuación expone los diferentes modelos ajustados, el valor del AIC y el p-valor obtenido del test de Ljung-Box. De haber pasado algún modelo la diagnosis, el seleccionado para realizar predicciones del volumen de ventas habría sido aquel con menor valor del AIC.

	MODELO	AIC	p-valor
Modelo 1	ARIMA(0,1,1)_7	1294.474	0.0025242
Modelo 2	ARIMA(1,0,0)x(2,1,0)_7	1328.403	0.0001488
Modelo 3	ARIMA(1,1,1)_7	1293.397	0.0068141
Modelo 4	ARIMA(1,1,0)_7	1341.251	0.0000002

Note:

El p-valor corresponde al test de Ljung-Box