

EJERCICIO SERIES TEMPORALES CLASE 2

Ejercicio 1. Sobre los datos de la moneda Bitcoin realizar los siguientes apartados sobre la variable precio:

1. Crear un objeto tipo zoo con los datos del fichero Bitcoin_A, generando previamente la secuencia temporal.

```
#Creamos la secuencia de dias en los que hemos observado la serie
dt= seq(from=as.Date("2018/09/18"),by="day", length.out=365)
head(dt)
```

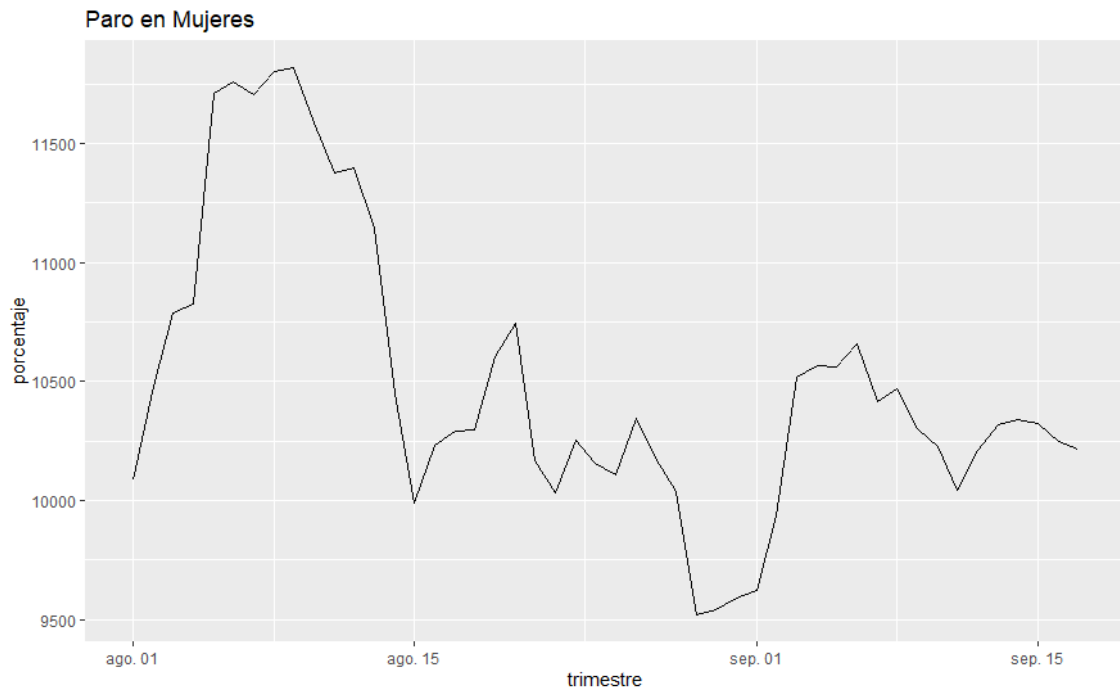
```
#Creamos los objetos tipo zoo que son otra forma de trabajar con series de tiempo
BITCOINz = zoo(x=bitcoin_A[,-1], order.by=dt)
precioBz = zoo(x=bitcoin_A[,2], order.by=dt)
ntransBz = zoo(x=bitcoin_A[,3], order.by=dt)
```

2. Crear una nueva serie precio_Rz con solo los datos del precio a partir del 1 de agosto de 2019

```
precio_Rz<-window(precioBz,start="2019/08/01")
print(precio_Rz)
```

3. Representar gráficamente la serie

```
autoplot(precio_Rz)+ ggtitle("Paro en Mujeres") + xlab("trimestre") + ylab("porcentaje")
```



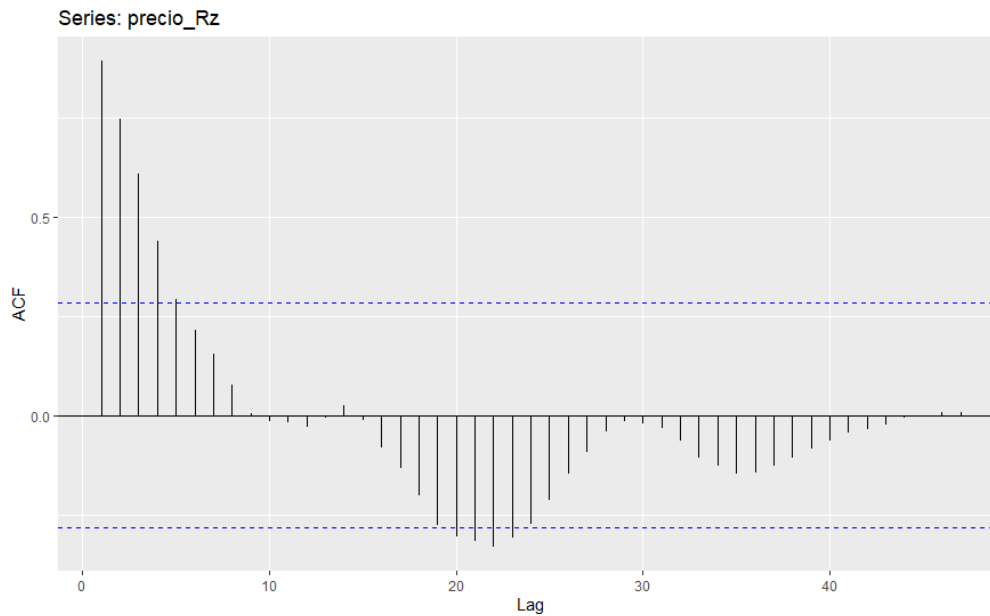
4. Representar los correlogramas simple y parcial. ¿Qué podemos decir a la vista del autocorrelograma simple? ¿Cuánto valen el coeficiente de autocorrelación simple de orden 1 y el de orden 5? ¿Y los correspondientes parciales ?

#Calculamos las autocorrelaciones simples hasta el retardo 48

```
ggAcf(precio_Rz, lag=48)
```

#Calculamos las autocorrelaciones parciales hasta el retardo 48

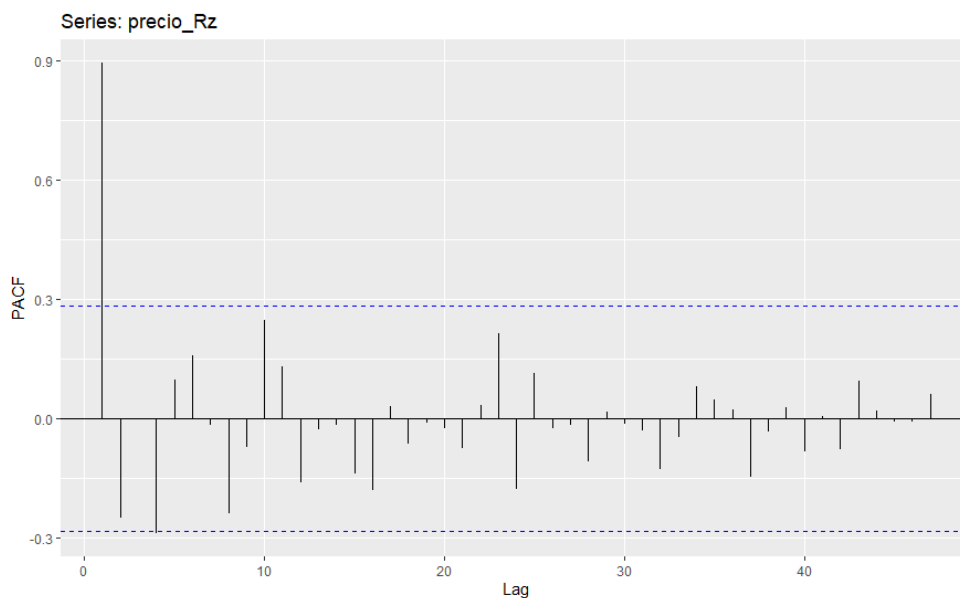
```
ggPacf(precio_Rz, lag=48)
```



```
corr<-acf(precio_Rz, lag=10)
```

```
print(corr)
```

	0	1	2	3	4	5	6	7	8	9	10
1.	0.892	0.745	0.608	0.438	0.292	0.213	0.155	0.075	0.004	-0.016	



```
corr<-Pacf(precio_Rz, lag=10)
```

```
print(corrp)
```

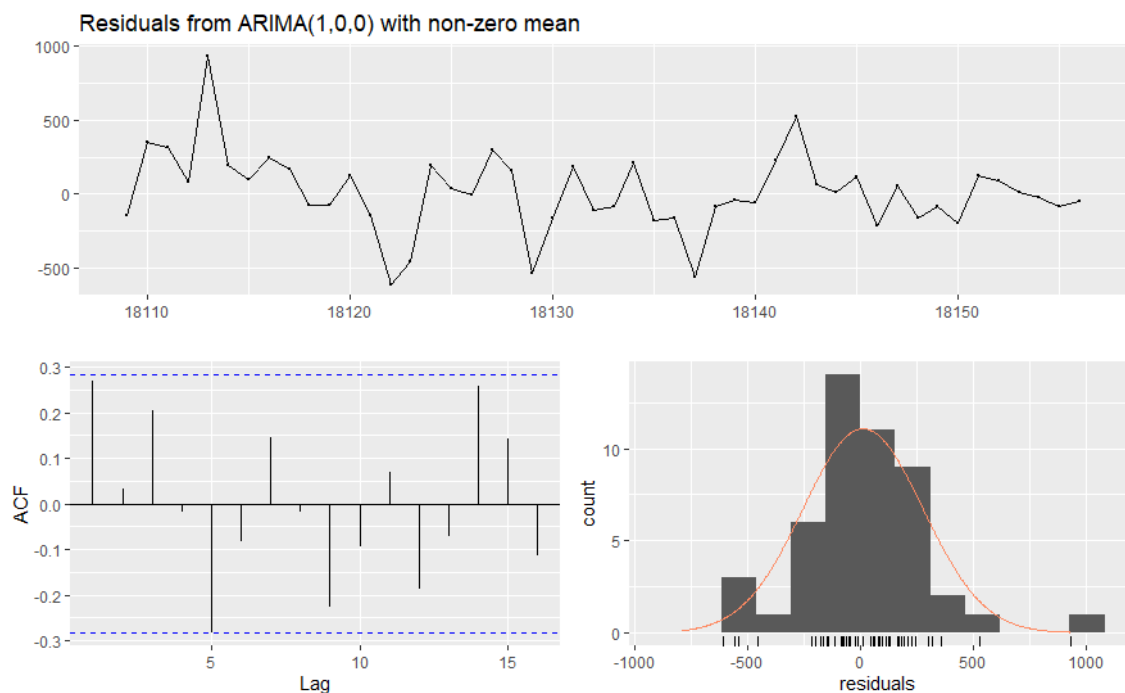
1	2	3	4	5	6	7	8	9	10
0.892	-0.250	0.000	-0.287	0.098	0.160	-0.017	-0.237	-0.072	0.247

5. Ajustar el modelo ARIMA adecuado mediante la función ARIMA y automáticamente, compararlos. ¿Los residuos están incorrelados?

```
#Ajuste del ARIMA(1,0,0)
```

```
fitprecio1 <- Arima(precio_Rz,c(1,0,0))
```

```
checkresiduals(fitprecio1)
```



Ljung-Box test

data: Residuals from ARIMA(1,0,0) with non-zero mean

Q* = 15.865, df = 8, p-value = 0.04435

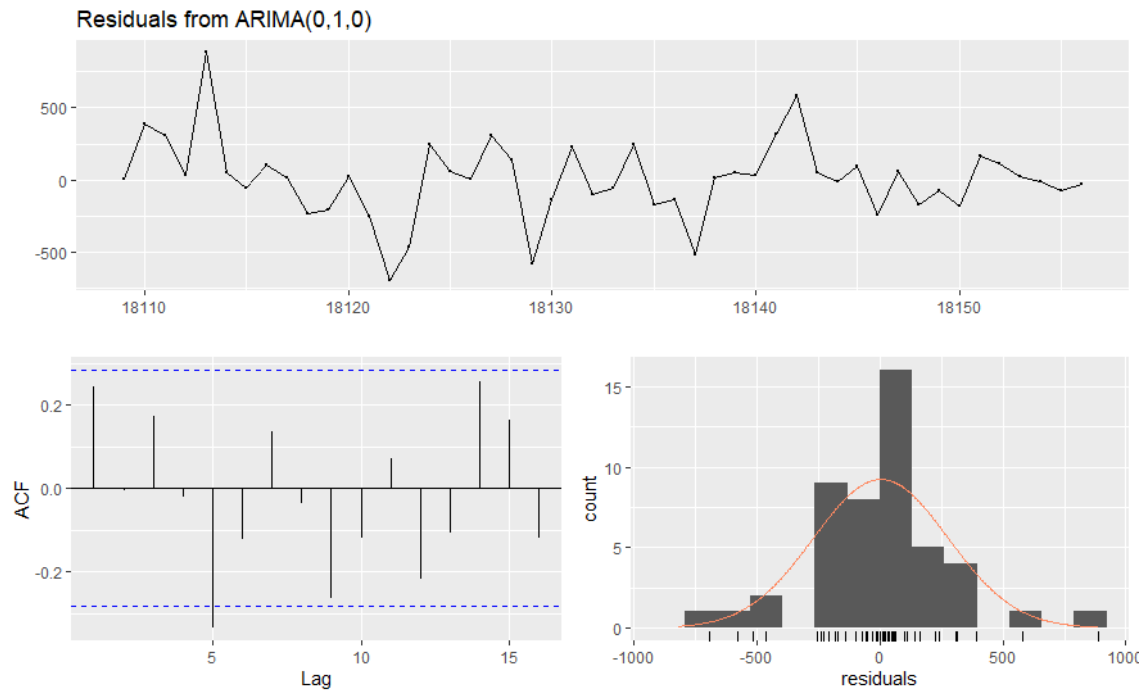
Model df: 2. Total lags used: 10

Con el p-valor rechazaríamos que los residuos están incorrelados a un nivel de significación de 0.05. Sin embargo, gráficamente vemos que las aoutocorrelaciones se puede aceptar que son cero puesto que están dentro de las bandas de confianza.

Utilizamos ahora la función de ajuste automático

```
fitprecio2 <- auto.arima(precio_Rz, seasonal=FALSE)
```

```
checkresiduals(fitprecio2)
```



Ljung-Box test

data: Residuals from ARIMA(0,1,0)
Q* = 18.149, df = 10, p-value = 0.0525
Model df: 0. Total lags used: 10

El pvalor es mayor pero es porque los grados de libertad han aumentado. Sin embargo las autocorrelaciones son mayores y alguna se sale fuera de las bandas de confianza.

6. Escribir la expresión teórica del mejor modelo con el valor de los estimadores

`print(fitprecio1)`

Series: precio_Rz
ARIMA(1,0,0) with non-zero mean

Coefficients:

	ar1	mean
	0.8882	10414.1591
s.e.	0.0589	298.0578

$(1 - 0.8882B)(X_t - 10414.15) = Z_t$

$$(1 - 0.8882B)(X_t - 10414.15) = Z_t$$

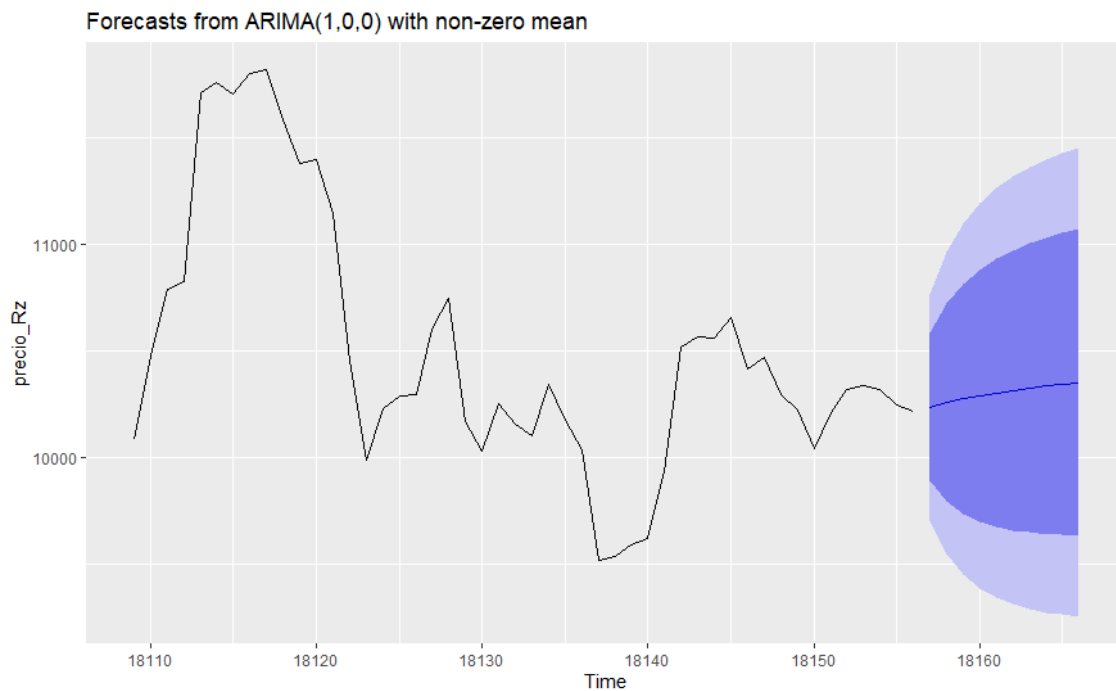
$$X_t - 0.8882X_{t-1} - 10414.15 + 0.8882 \cdot 10414.15 = Z_t$$

$$X_t = +1164.31 + 0.8882X_{t-1} + Z_t$$

sigma^2 estimated as 72825: log likelihood=-336.56
AIC=679.13 AICc=679.67 BIC=684.74

7. Calcular las predicciones para una semana

`autoplot(forecast(fitprecio1),h=7)`



`print(forecast(fitprecio1),h=7)`

Point	Forecast	Lo 80	Hi 80	Lo 95	Hi 95
18157	10239,22	9893,385	10585,07	9710,308	10768,14
18158	10258,78	9796,228	10721,34	9551,365	10966,21
18159	10276,16	9739,136	10813,18	9454,854	11097,46
18160	10291,59	9702,427	10880,75	9390,544	11192,63
18161	10305,29	9678,052	10932,54	9346,01	11264,58
18162	10317,47	9661,742	10973,19	9314,622	11320,31
18163	10328,28	9650,929	11005,63	9292,362	11364,19
18164	10337,88	9643,948	11031,81	9276,602	11399,16
18165	10346,41	9639,669	11053,15	9265,543	11427,28
18166	10353,98	9637,302	11070,67	9257,913	11450,06

Ejercicio 2.

Dada la serie, vuelos regulares en España desde Enero de 1995, con datos mensuales.

Utilizando la serie VuelosTR:

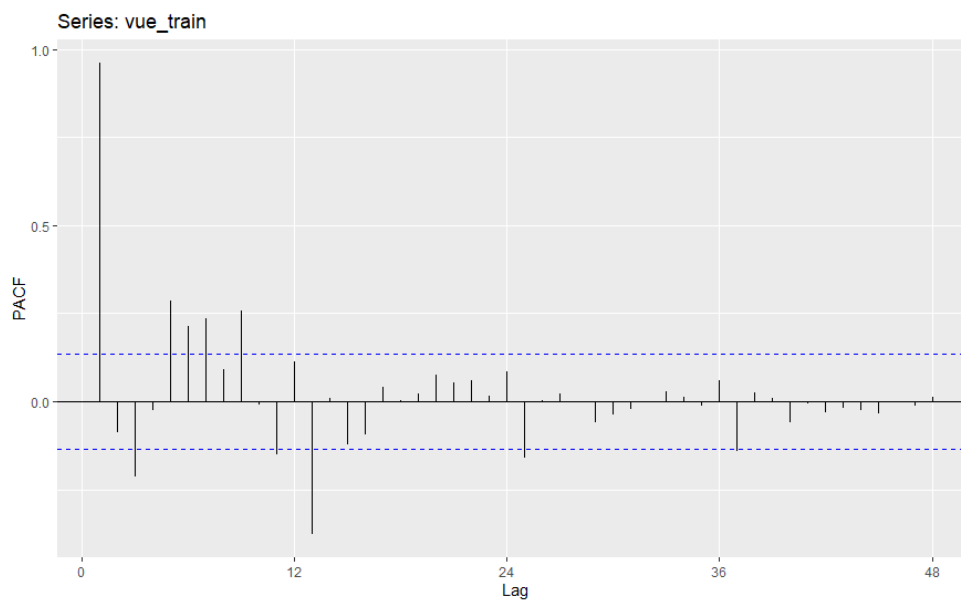
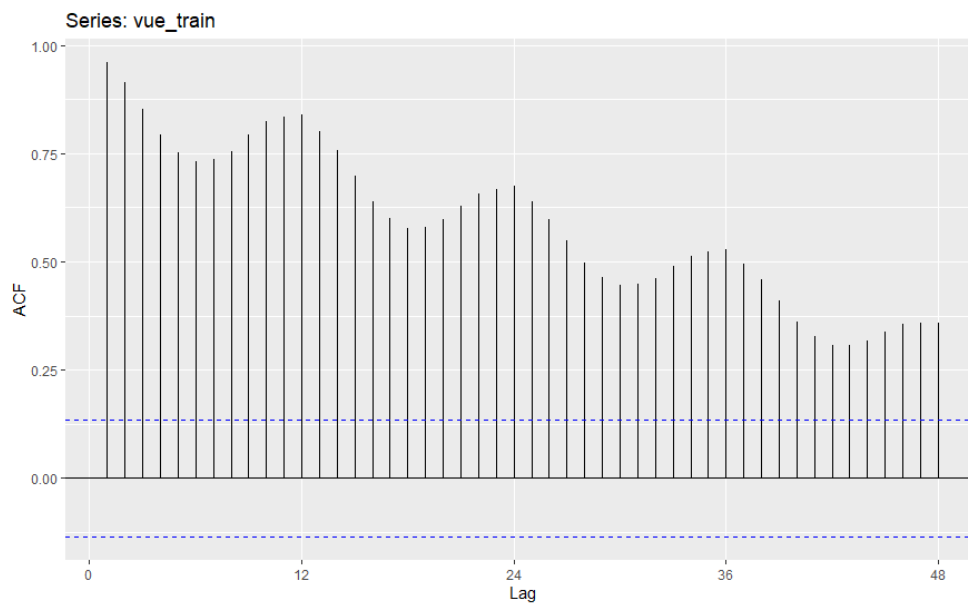
1. Representar la serie y los correlogramas

#Calculamos las autocorrelaciones simples hasta el retardo 48

```
ggAcf(vue_train, lag=48)
```

#Calculamos las autocorrelaciones parciales hasta el retardo 48

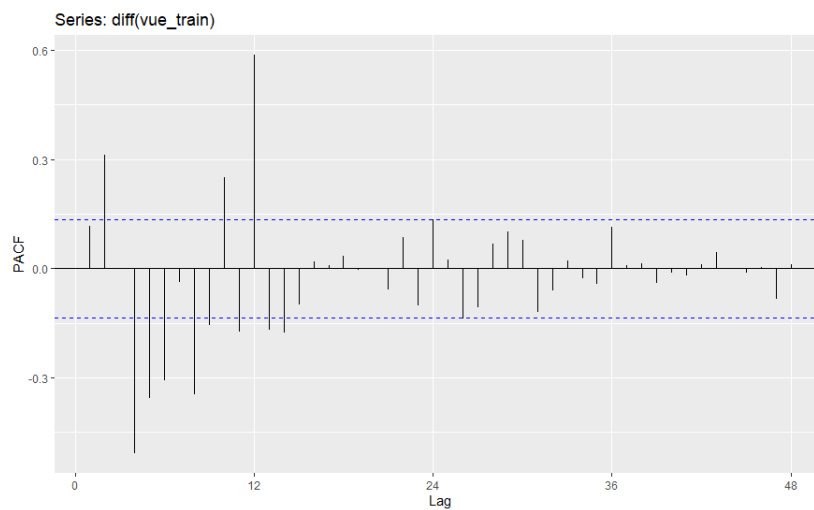
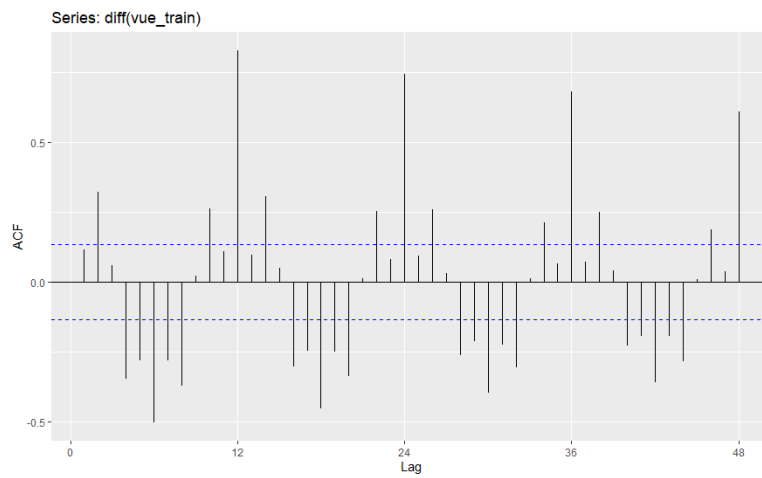
```
ggPacf(vue_train, lag=48)
```



2. Realizar las diferenciaciones de orden adecuado y volver a representar los autocorrelogramas

```
ggAcf(diff(vue_train), lag=48)
```

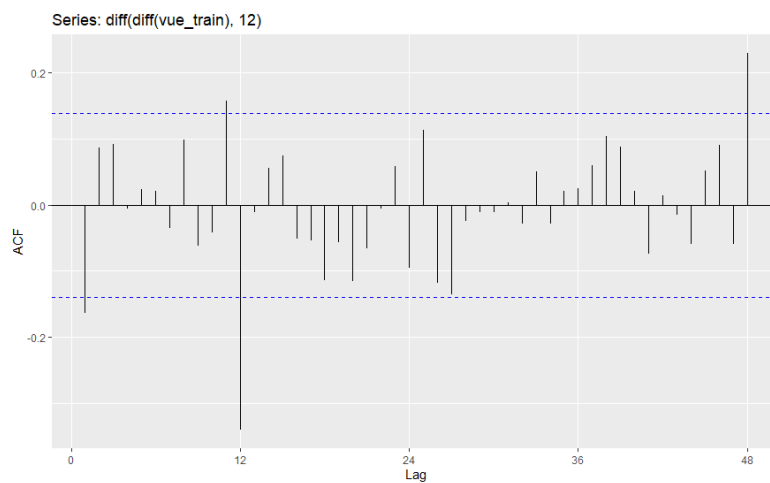
```
ggPacf(diff(vue_train), lag=48)
```

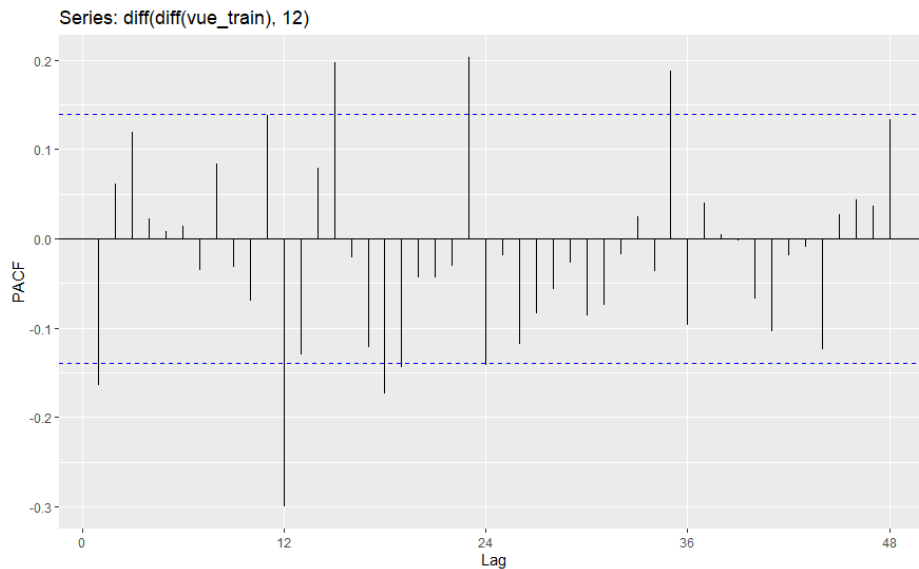


Hacemos una diferenciación de orden estacional

```
ggAcf(diff(diff(vue_train),12), lag=48)
```

```
ggPacf(diff(diff(vue_train),12), lag=48)
```





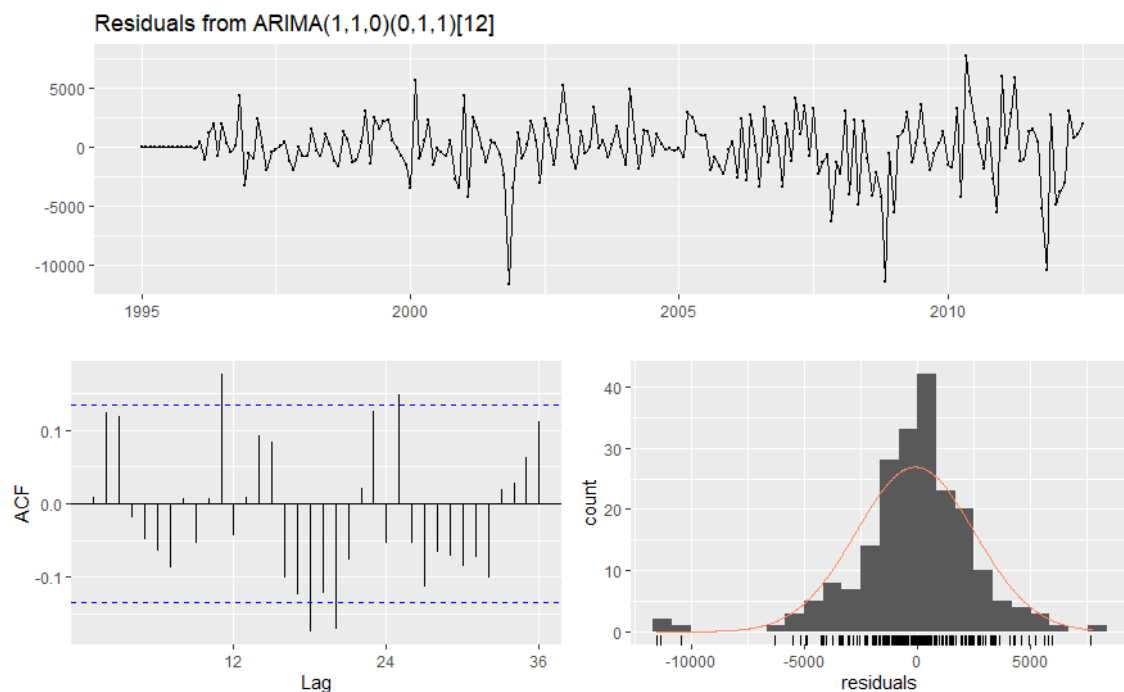
- Decidir que modelo puede ser ajustado. Ajustar el modelo adecuado comprobando que sus residuales están incorrelados.

Puesto que en los autocorrelogramas de la serie diferenciada se salen la barra 1 y 12 debemos que incluir estos retardos en el modelo.

#Ajuste del ARIMA(1,1,0)(0,1,1)

```
fitvuelos1 <- Arima((vue_train),c(1,1,0),seasonal=c(0,1,1))
```

```
checkresiduals(fitvuelos1)
```



Ljung-Box test

```
data: Residuals from ARIMA(1,1,0)(0,1,1)[12]
```

Q* = 50.382, df = 22, p-value = 0.0005206

Model df: 2. Total lags used: 24

Puesto que el pvalor es mucho menor de 0.05 rechazamos que los residuos están incorrelados, lo que implica que el modelo no explica toda la dependencia de la serie.

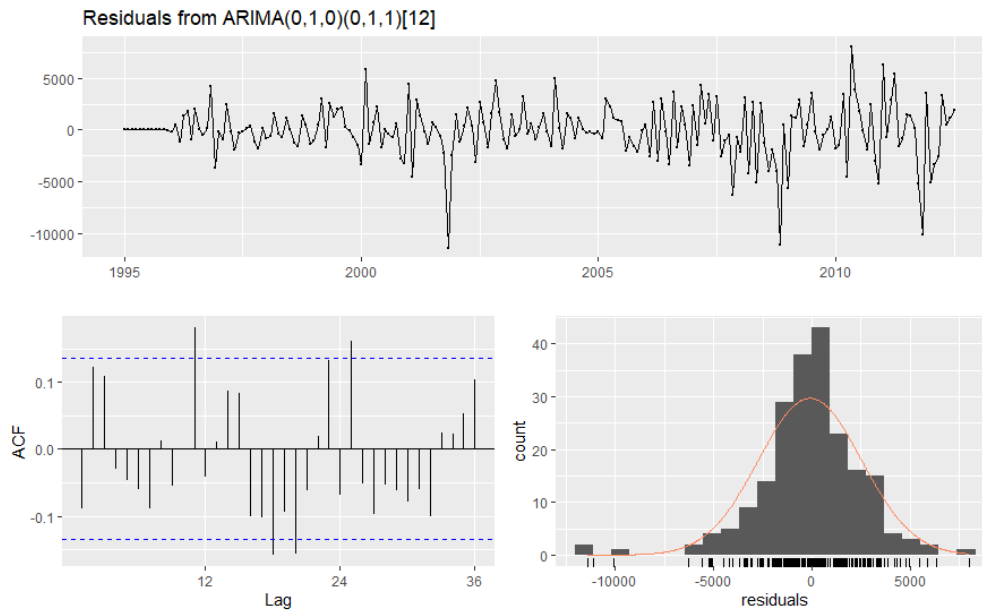
Probamos con la función autoArima para ver cual nos recomienda

```
fitvuelos2 <- auto.arima(vue_train)
checkresiduals(fitvuelos2)
autoplot(forecast(fitvuelos22),h=12)
```

Ljung-Box test

data: Residuals from ARIMA(0,1,0)(0,1,1)[12]
Q* = 47.114, df = 23, p-value = 0.002169

Model df: 1. Total lags used: 24



Como el pvalor sigue siendo menor de 0.05 probamos a tomar logaritmos porque inicialmente se observa que la variabilidad de la serie no permanece constante

#Ajuste mediante función autoARIMA sobre el logaritmo

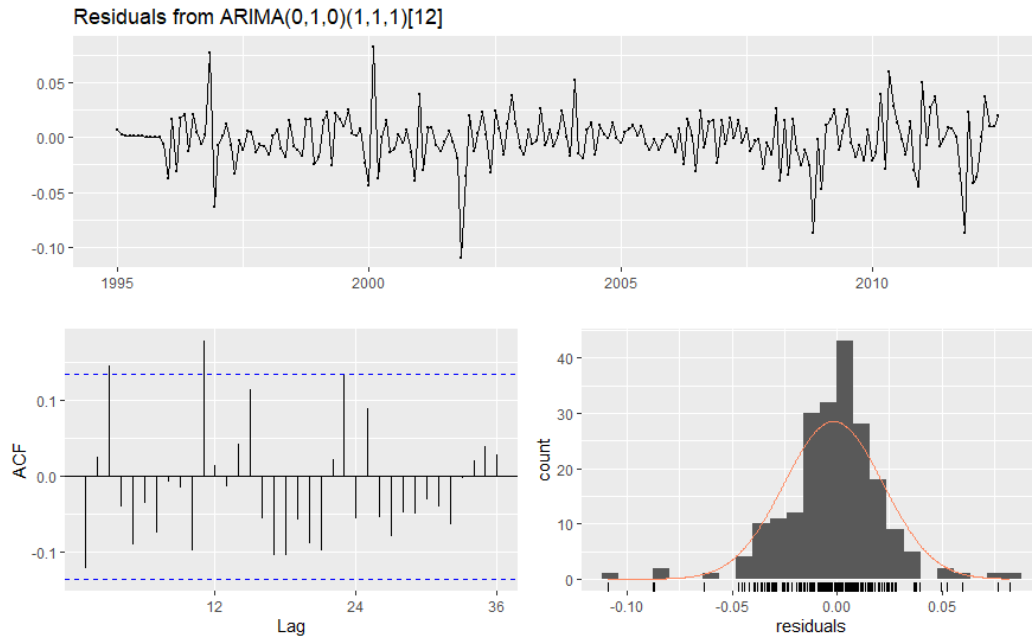
```
fitvuelos3 <- auto.arima(log(vue_train))
checkresiduals(fitvuelos3)
```

Ljung-Box test

data: Residuals from ARIMA(0,1,0)(1,1,1)[12]

$Q^* = 39.839$, $df = 22$, $p\text{-value} = 0.01129$

Model df: 2. Total lags used: 24



Este modelo puede valer porque el pvalor a nivel 0.01 acepta la hipótesis de que los residuos están incorrelados. Sin embargo en el grafico de as autocorrelaciones de los residuos vemos que sigue habiendo algun que no son cero especialmente la 11.

4. Escribir la expresión algebraica del modelo ajustado con los parámetros estimados.

`print(fitvuelos3)`

Series: log(vue_train)
ARIMA(0,1,0)(1,1,1)[12]

Coefficients:

	sar1	sma1
	-0.0305	-0.5604
s.e.	0.1216	0.1024

sigma^2 estimated as 0.0005975: log likelihood=453.73
AIC=-901.46 AICC=-901.34 BIC=-891.6

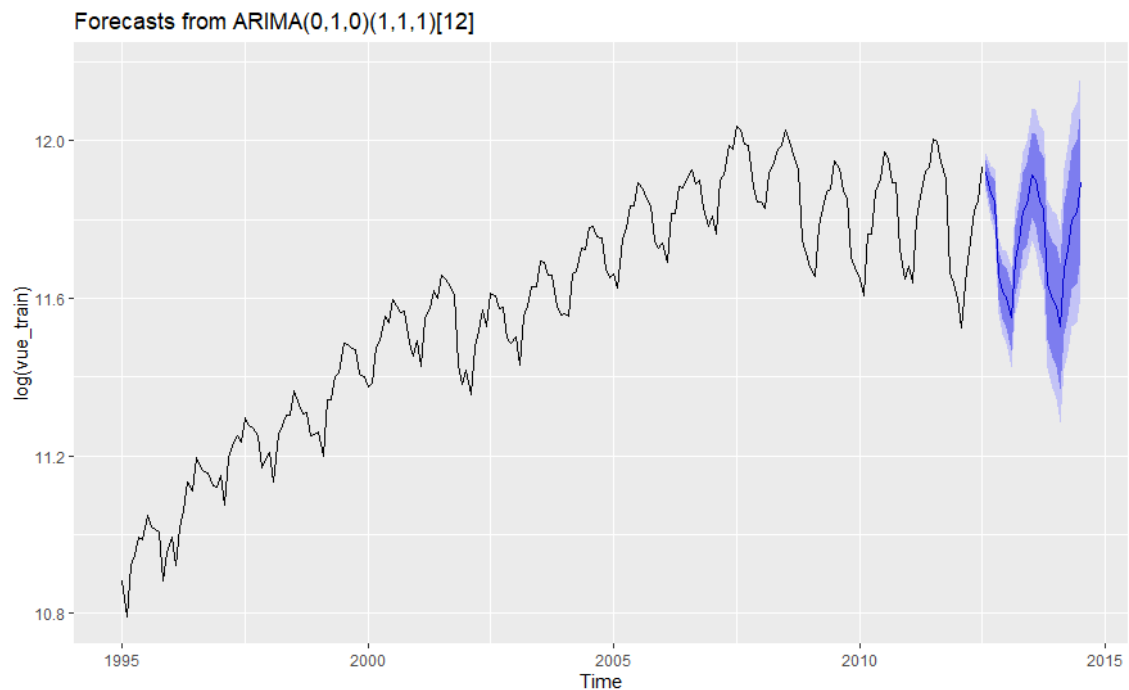
$$\begin{aligned}
 (1 + 0.03B^{12})(1 - B^{12})(1 - B)\log X_t &= (1 - 0.56B^{12})Z_t \\
 (1 - 0.03B^{12})(1 - B^{12})(\log X_t - \log X_{t-1}) &= -0.56Z_{t-12} + Z_t \\
 (1 - 0.03B^{12})(\log X_t - \log X_{t-12} + \log X_{t-1} - \log X_{t-13}) &= -0.56Z_{t-12} + Z_t \\
 \log X_t - \log X_{t-12} + \log X_{t-1} - \log X_{t-13} - 0.03\log X_{t-12} + 0.03\log X_{t-24} - 0.03\log X_{t-13} + 0.03\log X_{t-25} &= -0.56Z_{t-12} + Z_t \\
 \log X_t - 1.03\log X_{t-12} + \log X_{t-1} - 1.03\log X_{t-13} + 0.03\log X_{t-24} + 0.03\log X_{t-25} &= -0.56Z_{t-12} + Z_t \\
 \log X_t &= 1.03\log X_{t-12} - \log X_{t-1} + 1.03\log X_{t-13} - 0.03\log X_{t-24} - 0.03\log X_{t-25} - 0.56Z_{t-12} + Z_t
 \end{aligned}$$

Tomamos exponenciales para volver a los valores iniciales

$$X_t = X_{t-12}^{1.03} \cdot X_{t-13}^{1.03} \cdot X_{t-1}^{-1} \cdot X_{t-24}^{-0.03} \cdot X_{t-25}^{-0.03} \cdot Z_{t-12}^{0.56} \cdot Z_t'$$

5. Calcular las predicciones y los intervalos de confianza para las unidades de tiempo que se considere oportuno, dependiendo de la serie, siguientes al último valor observado. Mostradas en una tabla y representarlas gráficamente.

```
autoplot(forecast(fitvuelos3),h=12)
```



Observemos que los valores representados en el gráfico son el logaritmo de las predicciones.

Para obtener las predicciones reales, sin el logaritmo, tomamos exponenciales.

```
predi3<-forecast(fitvuelos3)
cbind("prediccion" =exp(predi3$mean),
      "L" = exp(predi3$lower),
      "U" = exp(predi3$upper)) %>%print()
```

		prediccion	L,80%	L,95%	U,80%	U,95%
Aug	2012	150146,3	145515,56	143122,29	154924,4	157515
Sep	2012	142581,3	136402,34	133240,56	149040,1	152576,8
Oct	2012	139337,7	131978,7	128241,73	147107,1	151393,8
Nov	2012	115558,8	108540,69	104999,75	123030,6	127179,6
Dec	2012	110959,2	103452,52	99686,55	119010,5	123506,5
Jan	2013	109380,5	101301,1	97268,59	118104,3	123000,6
Feb	2013	103806,2	95549,28	91447,61	112776,7	117835
Mar	2013	120209,1	110016,09	104974,9	131346,6	137654,2
Apr	2013	126340,5	115007,71	109426	138789,9	145869,5
May	2013	135957,2	123134,24	116843,26	150115,5	158197,9
Jun	2013	138488	124821,34	118141,38	153651	162338,7
Jul	2013	149192,4	133849,61	126377,03	166294	176126,8
Aug	2013	147282,5	130999,96	123122,35	165588,9	176183,6
Sep	2013	139865,1	123406,19	115492,55	158519,1	169380,9
Oct	2013	136600,7	119618,82	111501,22	155993,5	167350,2
Nov	2013	113113,3	98345,82	91325,65	130098,2	140098,8
Dec	2013	108657,9	93831,9	86820,86	125826,5	135987,3
Jan	2014	107028,8	91826,4	84673,44	124747,9	135286,3
Feb	2014	101508,2	86548,77	79543,99	119053,2	129537,3
Mar	2014	117546,6	99624,15	91270,82	138693,3	151386,9
Apr	2014	123621,1	104167,72	95141,41	146707,5	160626
May	2014	133056,1	111491,92	101529,56	158791,1	174372,1
Jun	2014	135558,1	112973,58	102583,67	162657,4	179131,7
Jul	2014	146087,4	121108,68	109664,21	176217,9	194607,9

Observemos que aunque hemos dicho que calcule predicciones para un año (h=12) lo ha hecho para dos.

6. Comparar las predicciones obtenidas con cada uno de los métodos con los valores observados que habíamos .reservado antes.

```
autoplot(vuelos) +
  autolayer(forecast(fitvuelos1), series="manual", PI=FALSE) +
  autolayer(forecast(fitvuelos2), series="automatico", PI=FALSE) +
  autolayer(exp(predi3$mean), series="Logaritmos")+
  ggtitle("Predicciones por diferentes modelos ") + xlab("mes") +
  ylab("numero") +
  guides(colour=guide_legend(title="Forecast"))
```

