ASM - Time Series Project

Authors: Dmitriy Chukhray, Julian Fransen

Date: 2025-01-06

Contents

Title page (julian will do this later when we finish)

Table of contents (julian)

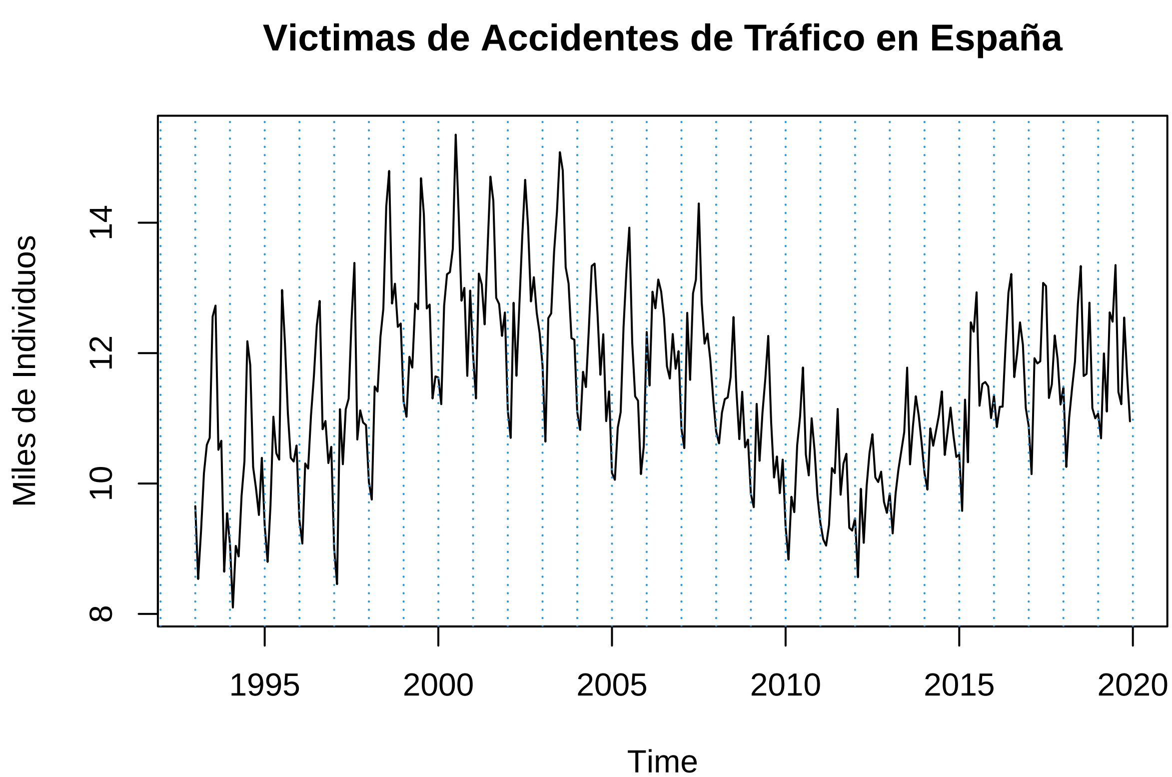
Introduction

The dataset under analysis consists of monthly data on victims of traffic accidents in Spain, including fatalities, serious injuries, and minor injuries, recorded on urban and interurban roads. In this project we will apply the Box-Jenkins ARIMA methodology to understand the time-series dynamics of these traffic incidents and to make reliable predictions for future trends. Spanning from 1993 to 2019, the dataset captures over two decades of detailed information, offering a unique opportunity to identify trends, seasonal patterns, and underlying factors that influence traffic accidents.

Analysis

Load the time series data

```
serie=ts(read.table("victimas.dat")/1000,start=1993,freq=12)
plot(serie, main="Victimas de Accidentes de Tráfico en España", ylab="Miles de Individuos")
abline(v=1992:2020,lty=3,col=4)
```



Data transformations

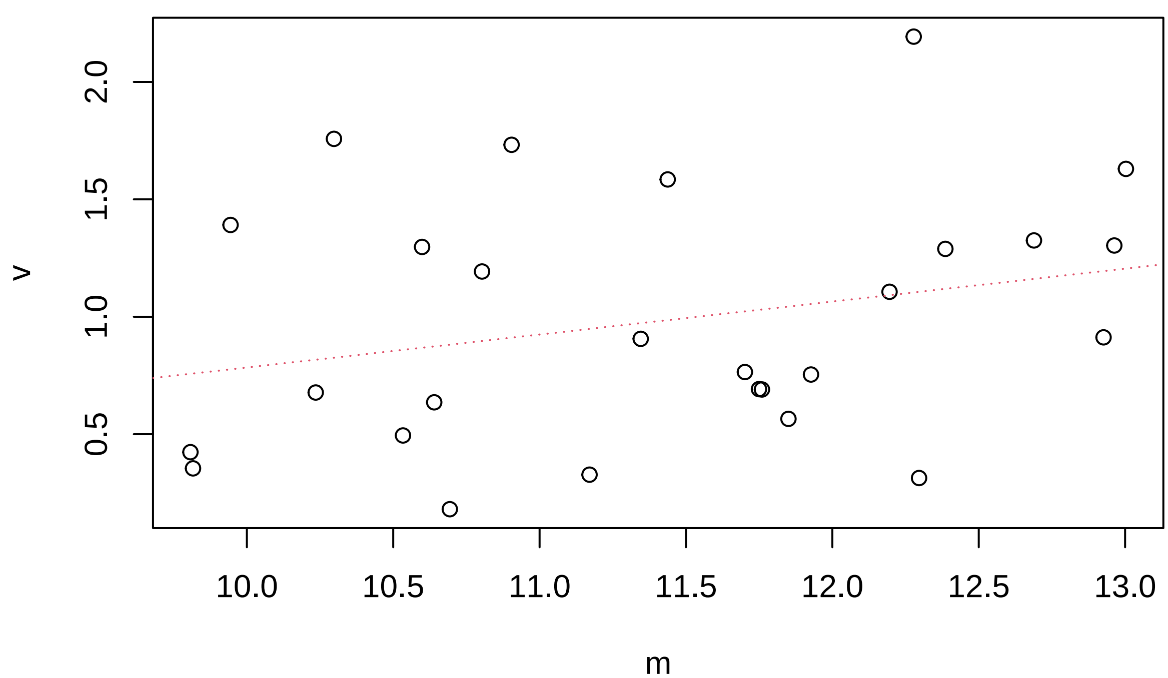
```r
(m <- apply (matrix(serie[1:(27*12)], nr=12),2, mean)) # to check for constant variance
```

```
##  [1] 10.297583  9.944250 10.598583 10.803333 10.904250 12.277833 12.386000
##  [8] 12.963083 12.926333 12.688667 13.002833 11.927000 11.437583 12.296167
## [15] 12.195333 11.170583 10.640000 10.235250  9.807250  9.816083 10.533333
## [22] 10.693333 11.345333 11.850000 11.749333 11.701250 11.759417
```

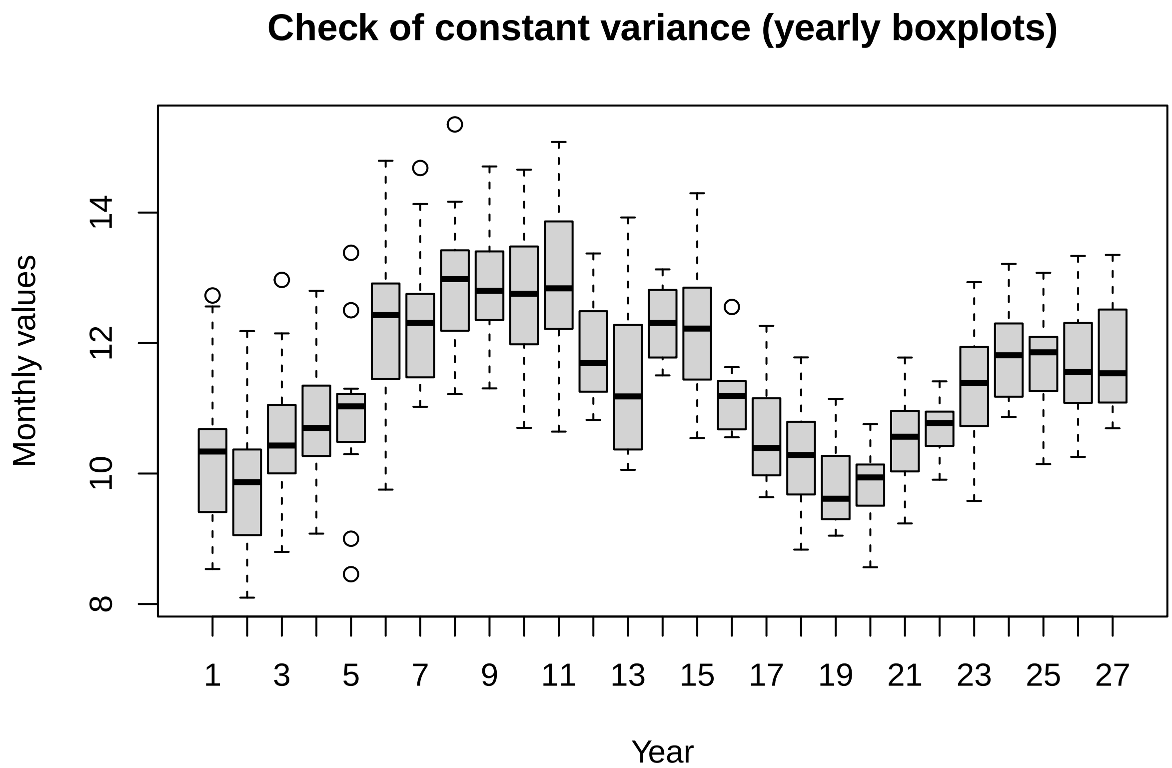
```r
(v <- apply (matrix(serie[1:(27*12)], nr=12),2, var))
```

```
##  [1] 1.7576637 1.3909389 1.2974164 1.1927470 1.7323289 2.1928725 1.2890669
##  [8] 1.3035081 0.9123124 1.3250108 1.6297787 0.7540431 1.5848150 0.3132571
## [15] 1.1063644 0.3275686 0.6356509 0.6774633 0.4234700 0.3544121 0.4944710
## [22] 0.1805739 0.9059681 0.5652725 0.6919446 0.7645724 0.6904190
```

```r
plot(v~m)
abline(lm(v~m),col=2,lty=3)
```

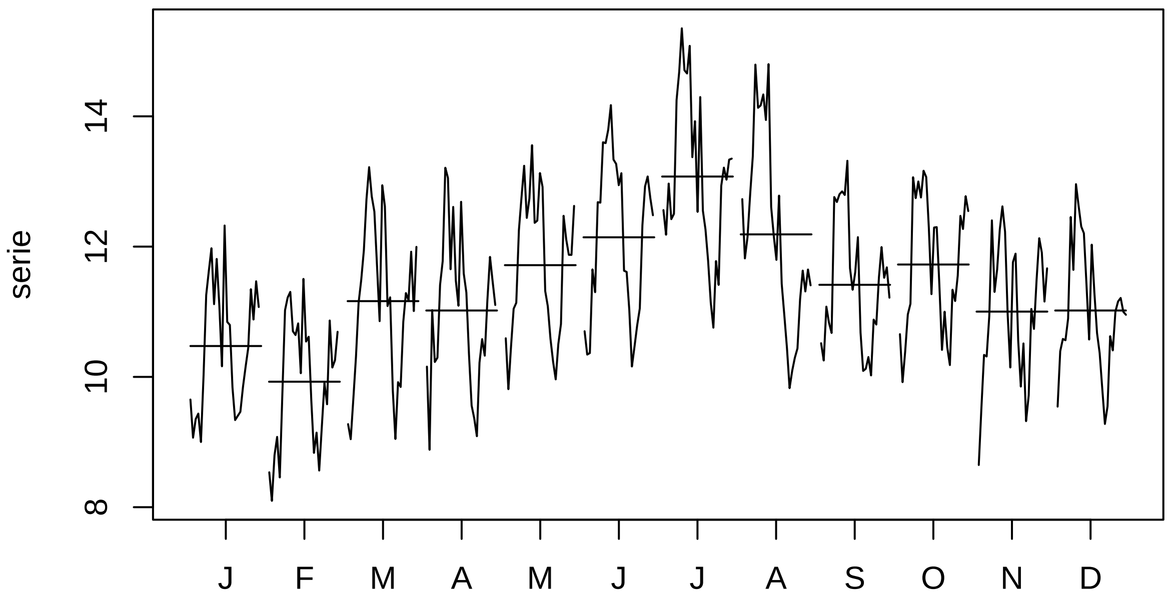



```r
group <- c(rep(1:27, rep(12, 27)))
boxplot(serie ~ group,
        xlab = "Year",
        ylab = "Monthly values",
        main = "Check of constant variance (yearly boxplots)")
```



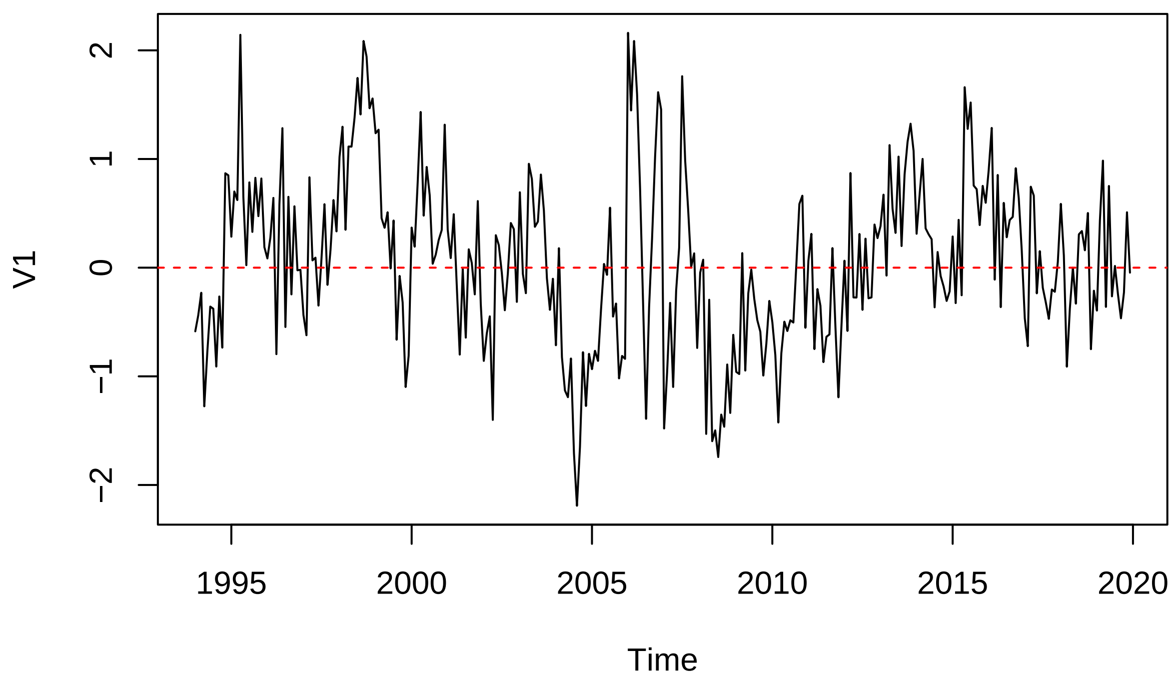
These plots show correct behaviour: `v` is basically uncorrelated with `m`, and the boxplots are similarily sized, implying constant variance. This means that a log transform is not necessary in our case. The next step is to check the existence of a seasonal pattern in the time series. To do that we are using the function monthplot.

```
monthplot(serie)
```



In the plot above, we can clearly observe a seasonal pattern. If there were no seasonal component, the monthly means would remain at approximately the same level over time, and the shapes of the patterns for each month would not exhibit systematic repetition. These periodic fluctuations suggest that certain months consistently experience higher or lower values, driven by underlying seasonal component(s). To account for this seasonality, we apply a seasonal differencing transformation with a yearly period (12 months).

```
d12serie=diff(serie,12)
plot(d12serie)
abline(h=0, col='red', lty=2)
```



The last step of achieving the stationary of the time series is to check whether the mean is constant or not. This can be done by examining the plot of the current time series data or by straight forwardly applying regular difference and then examining the change of the time series' variance.

```
d1d12serie=diff(d12serie,1)
```

To verify if taking one regular difference is optimal, we calculate the variance of different transformation of the data: 1. original data (`serie`) 2. transformed with yearly season transformation (`d12serie`) 3. one regular difference applied to the previous transformation (`d1d12serie`) 4. another regular difference applied to the previous transformation

```
#fix this output
var(serie)
```

```
##          V1
## V1 1.863327
```

```
var(d12serie)
```

```
##           V1
## V1 0.6081206
```

```
var(d1d12serie) # our transformation
```

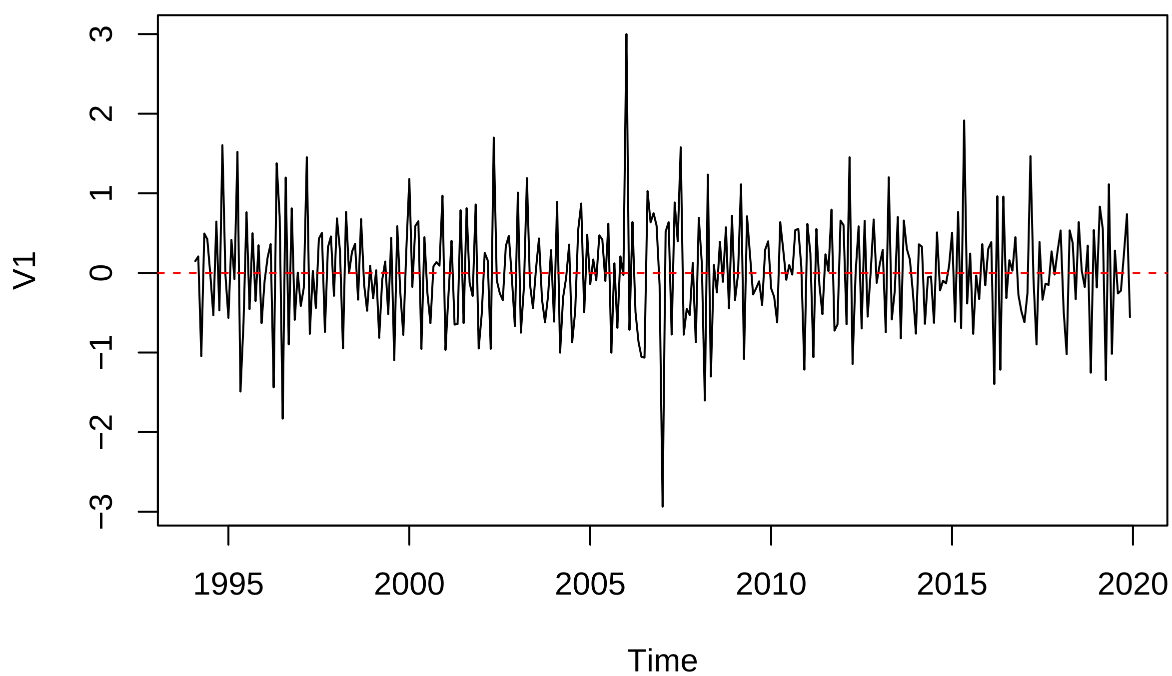
```
##           V1
## V1 0.4841846
```

```r
var(diff(d1d12serie)) # 2 regular differences
```

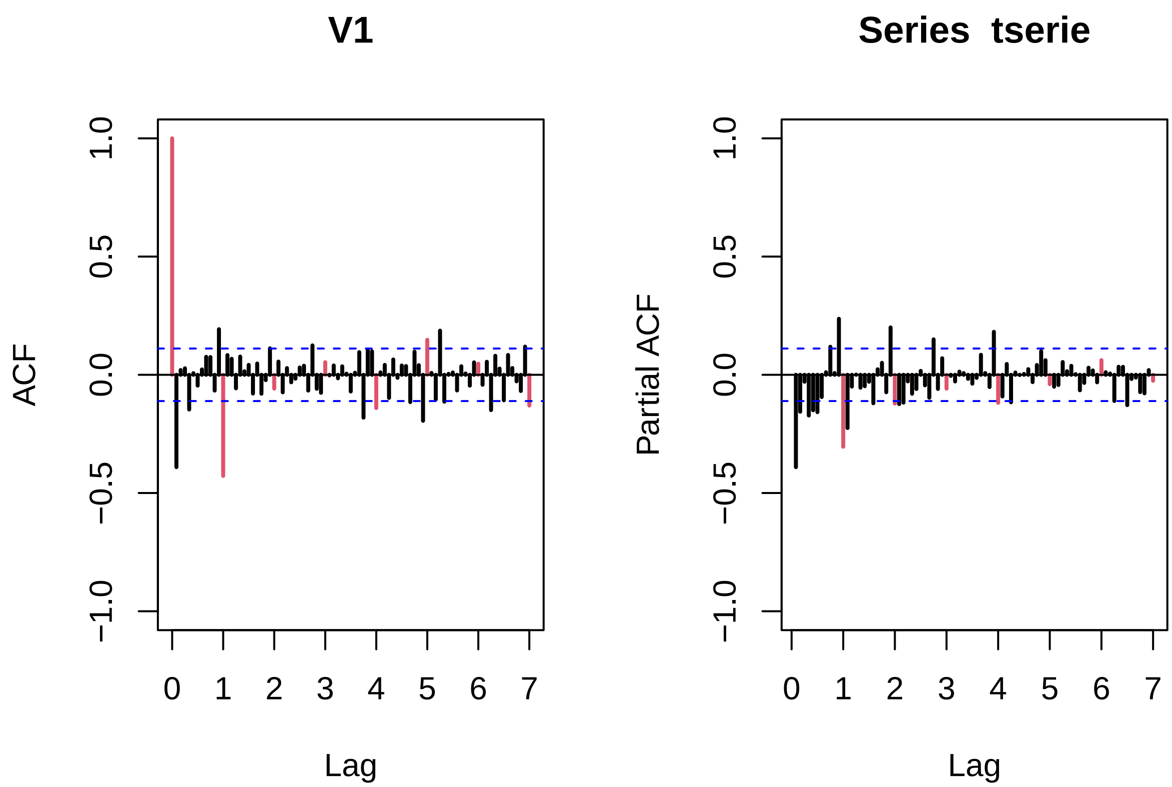
```
##          V1
## V1 1.349775
```

The total variance is minimal for one regular and one 12 month seasonal difference. This means that we should take one regular difference and no more. As we can see below, the data after transformations resembles white noise, which is the aim of data transformations in time-series data analysis. Now that we have stationary time series we can propose ARIMA models by evaluating ACF and PACF plots.

```r
plot(d1d12serie)
abline(h=0, col='red', lty=2)
```




```r
par(mfrow=c(1,2))
acf(tserie,ylim=c(-1,1),col=c(2,rep(1,11)),lwd=2,lag.max=84)
pacf(tserie,ylim=c(-1,1),col=c(rep(1,11),2),lwd=2,lag.max=84)
```



To propose SARIMA models we need to evaluate ACF and PACF plots separately. SARIMA model can be expressed as:

$$SARIMA(p, d, q)(P, D, Q)_s$$

Where: - p, d, q represent the non-seasonal AR, differencing, and MA terms, respectively. - P, D, Q represent the seasonal AR, differencing, and MA terms, respectively. - s is the periodicity of the seasonal component.

Looking at the ACF plot we can clearly see decaying trend of non-seasonal lags after the lag 1, which is the biggest lag for the whole non-seasonal part and the subsequent lag experienced a sharp decline in ACF value hinting at possible MA(1) model (parameter q = 1). The same things can be said about seasonal lags in the ACF plot. There is a decaying trend of seasonal lags after the seasonal lag 1, which is the biggest seasonal lag for the whole seasonal part and the subsequent seasonal lag experienced a sharp decline in ACF value hinting at possible MA(1) model (parameter Q = 1). Now we move on to the PACF plot, which looks more complex than usual. The clearer trend is seen in seasonal lags of the PACF. The first seasonal lag is the biggest seasonal lag for the whole seasonal part and the subsequent seasonal lag experienced a sharp decline in PACF value hinting at possible AR(1) model (parameter P = 1). Here this trend is clearer than in the case of seasonal lags in the ACF because after seasonal lag 4 values become small so that exceeding confidence intervals becomes nearly impossible. In the ACF plot we have seen some seasonal lags being slightly above confidence intervals even though their antecedent lags were way below confidence intervals. The proposal of non-seasonal AR models by solely looking at the PACF plot in this particular case seems non-trivial, hence it would be a good idea to propose several hypothetical non-seasonal AR models and later check them using several model validation techniques. The decaying trend of non-seasonal AR part is there, though it is slower and seems to start later comparing it to non-seasonal MA part. Therefore, one could argue that one possible non-seasonal AR model which is AR(2) (parameter p = 2) because lag 2 is followed a lag that has experienced a sharp decline in PACF value. However, lags 4,5, and 6 are definitely above confidence intervals. Here is where we can propose another non-seasonal AR mode which is AR(6) (parameter p = 6), because lag 6 is followed by lags that are below confidence intervals consequentially and the decline in their values is sharper. Regarding the values of parameters d and D, we don't need to propose them because we know the actual values which are d = 1 and D = 1. In the end, combining everything we have seen we can propose 9 SARIMA models in the order of increasing complexity, which are:

1. $SARIMA(0,1,1)(1,1,0)_{12}$ (Non-seasonal MA(1), Seasonal AR(1))

2. $SARIMA(0,1,1)(0,1,1)_{12}$ (Non-seasonal MA(1), Seasonal MA(1))

3. $SARIMA(0,1,1)(1,1,1)_{12}$ (Non-seasonal MA(1), Seasonal AR(1), Seasonal MA(1))

4. $SARIMA(2,1,1)(1,1,0)_{12}$ (Non-seasonal MA(1), Non-seasonal AR(2), Seasonal AR(1))

5. $SARIMA(2,1,1)(0,1,1)_{12}$ (Non-seasonal MA(1), Non-seasonal AR(2), Seasonal MA(1))

6. $SARIMA(2,1,1)(1,1,1)_{12}$ (Non-seasonal MA(1), Non-seasonal AR(2), Seasonal AR(1), Seasonal MA(1))

7. $SARIMA(6,1,1)(1,1,0)_{12}$ (Non-seasonal MA(1), Non-seasonal AR(6), Seasonal AR(1))

8. $SARIMA(6,1,1)(0,1,1)_{12}$ (Non-seasonal MA(1), Non-seasonal AR(6), Seasonal MA(1))

9. $SARIMA(6,1,1)(1,1,1)_{12}$ (Non-seasonal MA(1), Non-seasonal AR(6), Seasonal AR(1), Seasonal MA(1))

Validation

Now we will start model creation and validation. We use the `validation` function from the fourth practical of this course, as well as other pieces of code.

After experimenting with the `p`, `i`, and `q` parameters of `order`, and with the seasonality parameters as well, we have found 2 similarly performing models. `mod1` and `mod2` (see table below). Their value of `p` is different but all other parameters are identical. We chose these two models because they both have a low AIC value; `mod1` has the lowest and `mod2` has a slightly higher value, but has fewer parameters, reflected in the fact that is has the lowest BIC value, since this metric penalizes the number of parameters more than AIC.

```
##   p i q P D Q      AIC      BIC
## 2 0 1 1 0 1 1 473.8614 485.0808
## 6 2 1 1 0 1 1 471.1388 489.8378
```

```
##
## Call:
## arima(x = tserie, order = c(2, 0, 1), seasonal = list(order = c(0, 1, 1), period = 12))
##
## Coefficients:
##          ar1     ar2      ma1     sma1
##       0.4317  0.2182  -0.9643  -1.0000
## s.e.  0.0631  0.0615   0.0300   0.0311
##
## sigma^2 estimated as 0.3817:  log likelihood = -301.34,  aic = 612.68
```

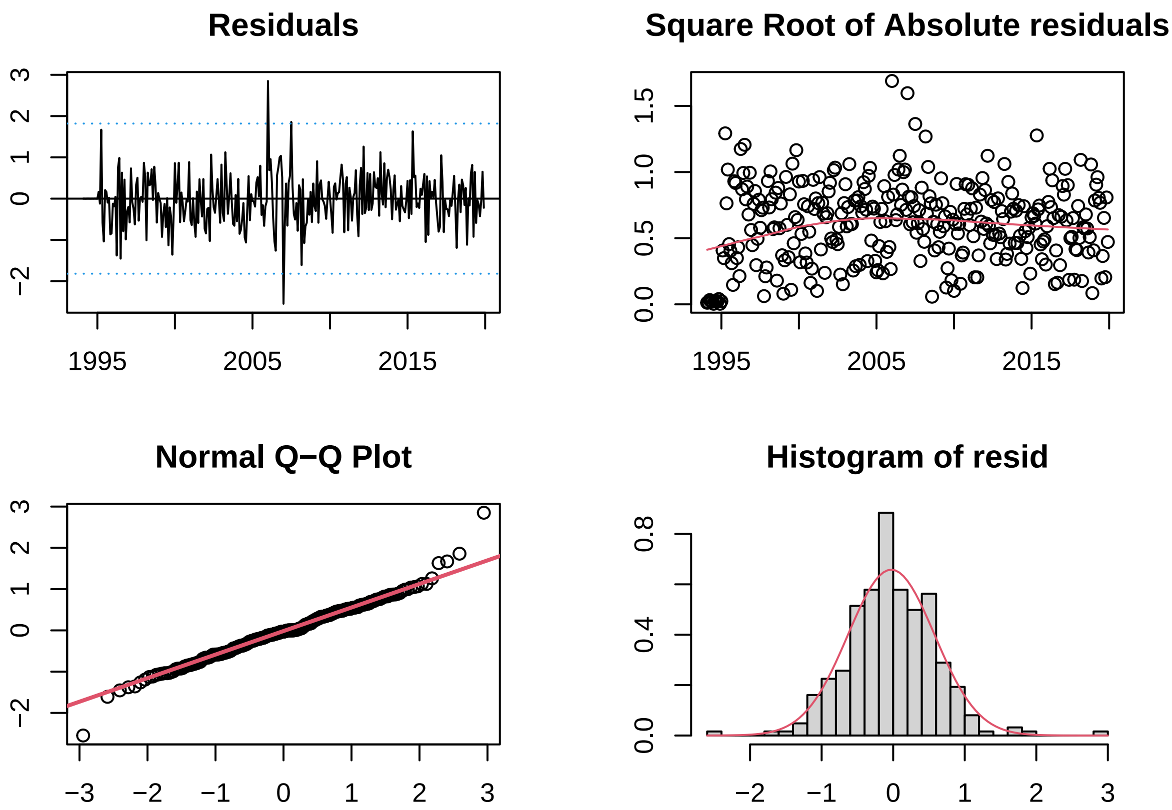
```
##
## Call:
## arima(x = tserie, order = c(0, 0, 1), seasonal = list(order = c(0, 1, 1), period = 12))
##
## Coefficients:
##           ma1     sma1
##       -0.5268  -1.0000
## s.e.   0.0619   0.0301
##
## sigma^2 estimated as 0.4044:  log likelihood = -308.62,  aic = 623.24
```

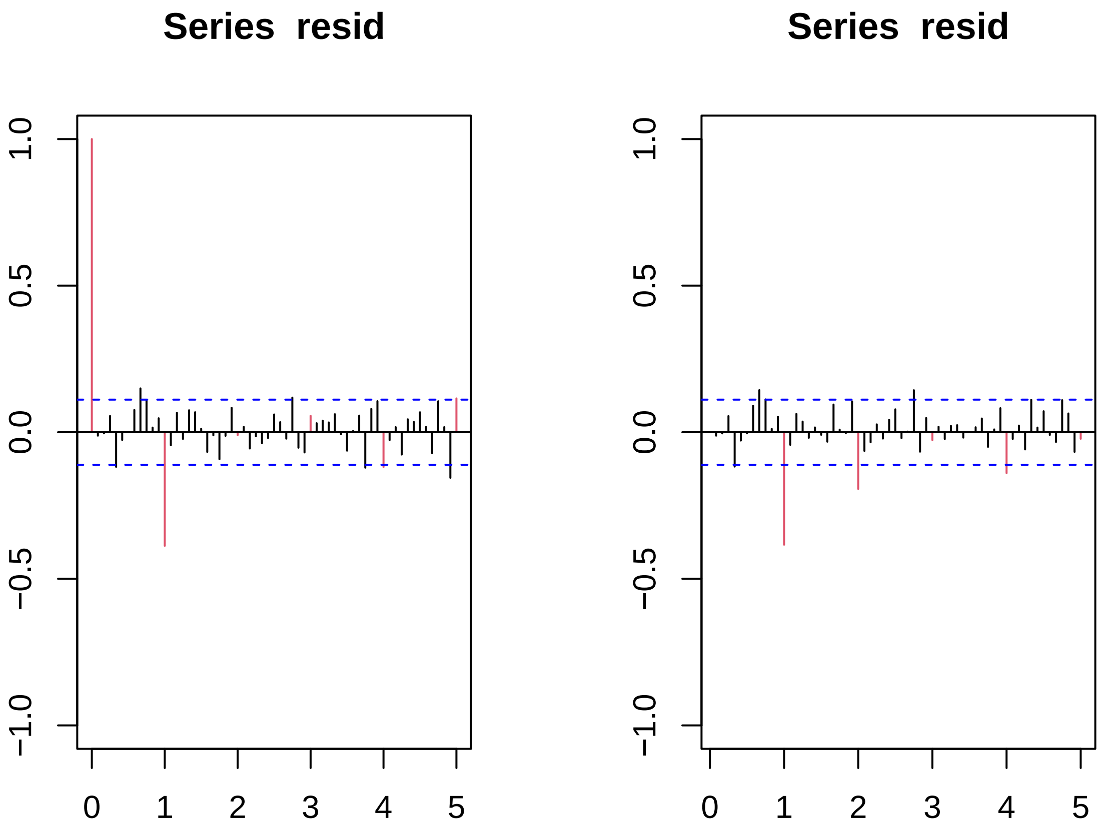
okay dima, from here you go go

use echo=FALSE to not show the code; use include to not show the code AND output, use "'plaintext for text boxes. I will take care of the title page, table of contents etc.

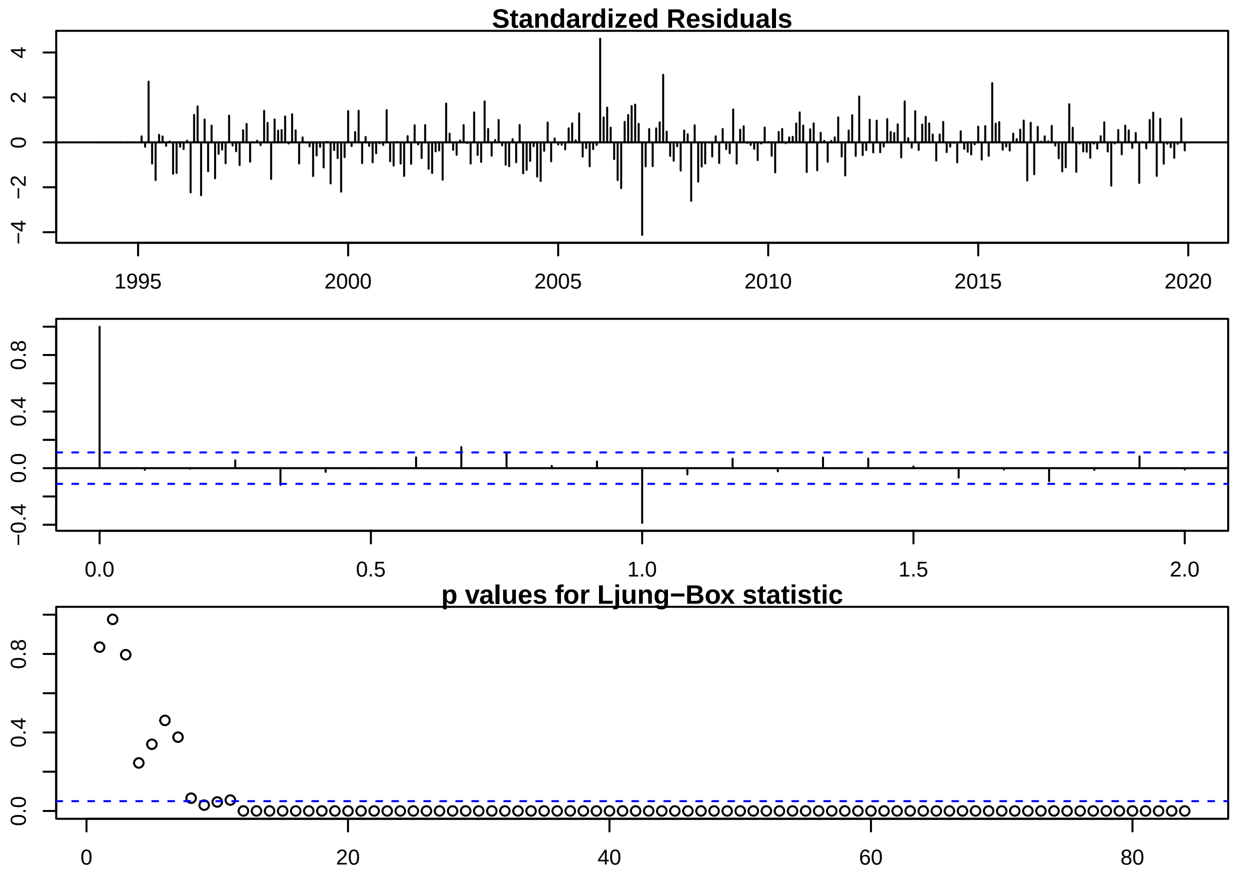
We will be taking a look at the residual plots for these models and selecting one for prediction.

```
validation(mod1)
```

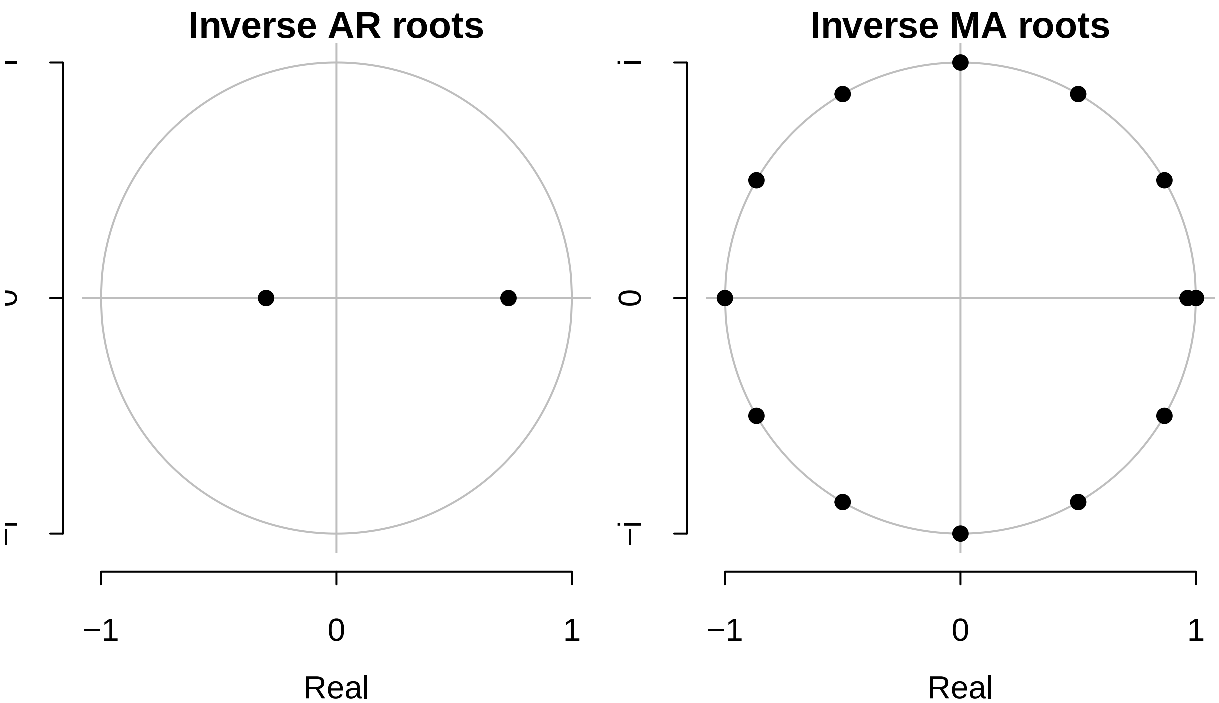

Residuals
Square Root of Absolute residuals
Normal Q−Q Plot
Histogram of resid



Series resid
1.0
0.5
0.0
−0.5
−1.0
0
1
2
3
4
5
Series resid
1.0
0.5
0.0
−0.5
−1.0
0
1
2
3
4
5




```
##
## --------------------------------------------------------------------
##
## Call:
## arima(x = tserie, order = c(2, 0, 1), seasonal = list(order = c(0, 1, 1), period = 12))
##
## Coefficients:
##          ar1     ar2      ma1     sma1
##       0.4317  0.2182  -0.9643  -1.0000
## s.e.  0.0631  0.0615   0.0300   0.0311
##
## sigma^2 estimated as 0.3817:  log likelihood = -301.34,  aic = 612.68
##
## Modul of AR Characteristic polynomial Roots:  1.369155 3.347821
##
## Modul of MA Characteristic polynomial Roots:  1.000001 1.000001 1.000001 1.000001 1.000001 1.000001
```

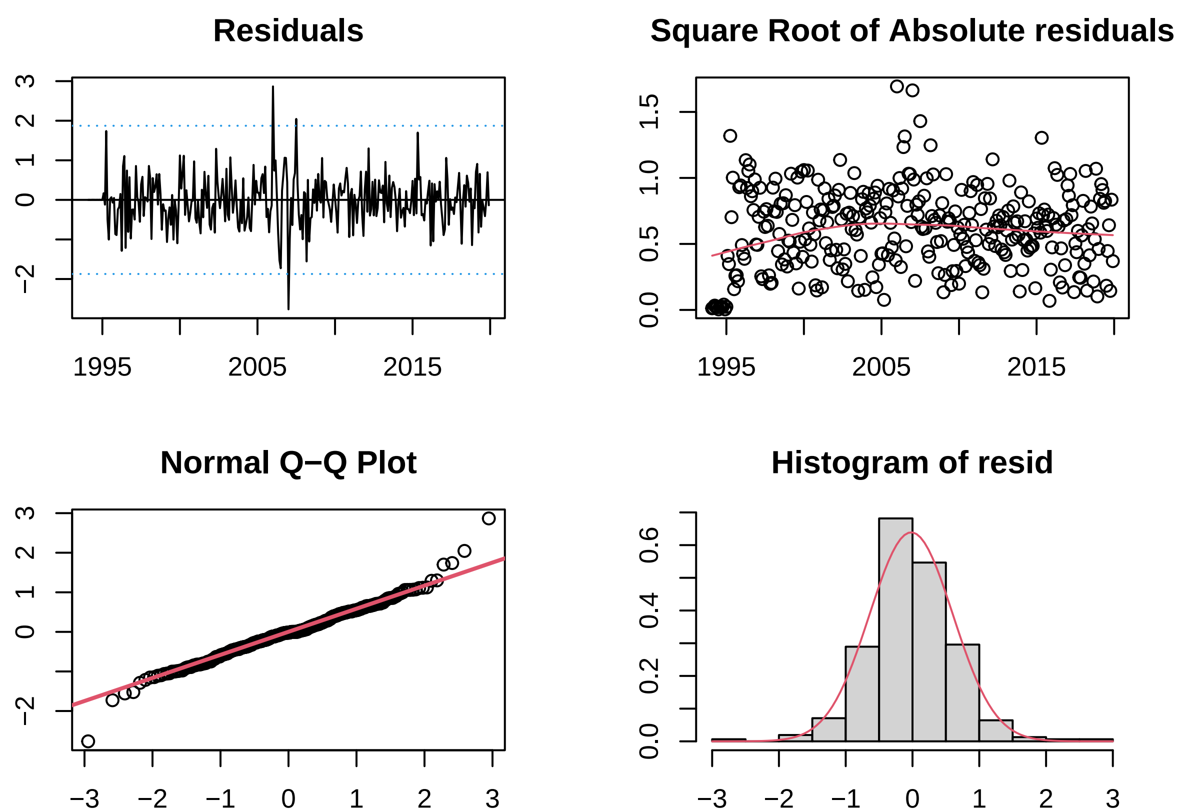


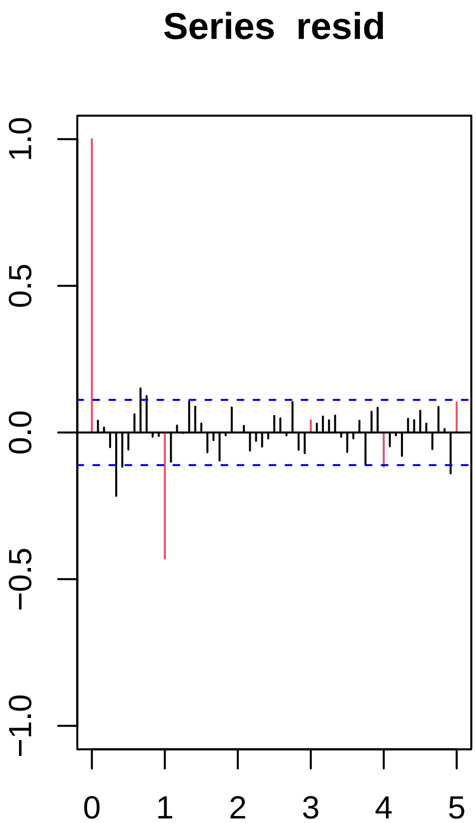

```
##
## Psi-weights (MA(inf))
##
## --------------------
##        psi 1        psi 2        psi 3        psi 4        psi 5        psi 6
## -0.532661278 -0.011771852 -0.121289646 -0.054925972 -0.050171346 -0.033640662
##        psi 7        psi 8        psi 9       psi 10       psi 11       psi 12
## -0.025467479 -0.018332900 -0.013469975 -0.009814254 -0.007175249 -1.005228643
##       psi 13       psi 14       psi 15       psi 16       psi 17       psi 18
##  0.528829302  0.008976975  0.119247164  0.053434538  0.049081932  0.032845010
##       psi 19       psi 20       psi 21       psi 22       psi 23       psi 24
##  0.024886343  0.017908455  0.013159969  0.009587833  0.007009876  0.005117721
##
## Pi-weights (AR(inf))
##
## --------------------
##       pi 1       pi 2       pi 3       pi 4       pi 5       pi 6       pi 7
## -0.5326613 -0.2954999 -0.2849614 -0.2747988 -0.2649985 -0.2555478 -0.2464342
##       pi 8       pi 9      pi 10      pi 11      pi 12      pi 13      pi 14
## -0.2376455 -0.2291703 -0.2209974 -0.2131159 -1.2055056 -0.7308421 -0.4866151
##      pi 15      pi 16      pi 17      pi 18      pi 19      pi 20      pi 21
## -0.4692609 -0.4525255 -0.4363870 -0.4208240 -0.4058160 -0.3913433 -0.3773868
##      pi 22      pi 23      pi 24
## -0.3639279 -0.3509491 -1.3384134
##
## Descriptive Statistics for the Residuals
```

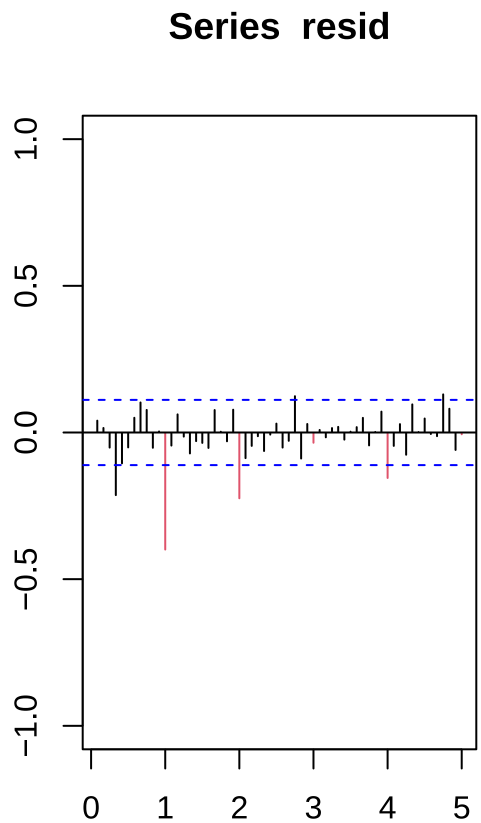
```
##
## ----------------------------------------
##                   resid
## nobs        311.000000
## NAs           0.000000
## Minimum      -2.547233
## Maximum       2.850616
## 1. Quartile  -0.401036
## 3. Quartile   0.368768
## Mean         -0.026851
## Median       -0.031149
## Sum          -8.350730
## SE Mean       0.034373
## LCL Mean     -0.094485
## UCL Mean      0.040782
## Variance      0.367442
## Stdev         0.606170
## Skewness      0.167149
## Kurtosis      1.986592
##
## Normality Tests
##
## --------------------
##
##  Shapiro-Wilk normality test
##
## data:  resid
## W = 0.98076, p-value = 0.0003458
##
##
##  Anderson-Darling normality test
##
## data:  resid
## A = 0.71837, p-value = 0.06034
##
##
##  Jarque Bera Test
##
## data:  resid
## X-squared = 54.275, df = 2, p-value = 1.638e-12
##
##
## Homoscedasticity Test
##
## --------------------
##
##  studentized Breusch-Pagan test
##
## data:  resid ~ I(obs - resid)
## BP = 0.16466, df = 1, p-value = 0.6849
##
##
## Independence Tests
##
```

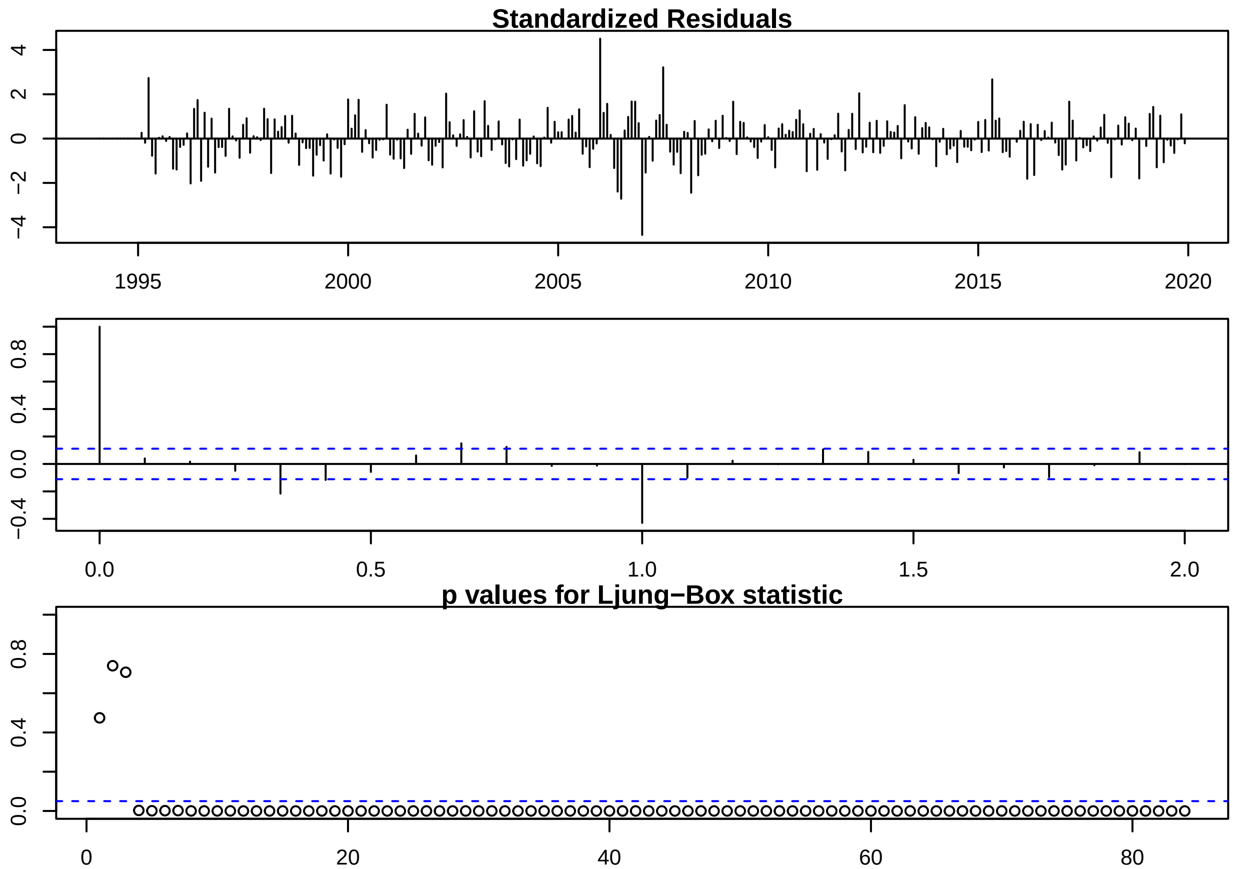
```
## --------------------
##
##  Durbin-Watson test
##
## data:  resid ~ I(1:length(resid))
## DW = 2.0346, p-value = 0.5982
## alternative hypothesis: true autocorrelation is greater than 0
##
##
## Ljung-Box test
##      lag.df    statistic      p.value
## [1,]      1   0.04342742 8.349225e-01
## [2,]      2   0.04807377 9.762497e-01
## [3,]      3   1.02174222 7.959912e-01
## [4,]      4   5.44528362 2.445811e-01
## [5,]     12  68.17940680 7.006039e-10
## [6,]     24  80.74382492 4.629194e-08
## [7,]     36  93.10269381 5.977437e-07
## [8,]     48 115.44426012 1.750539e-07
```

```
validation(mod2)
```

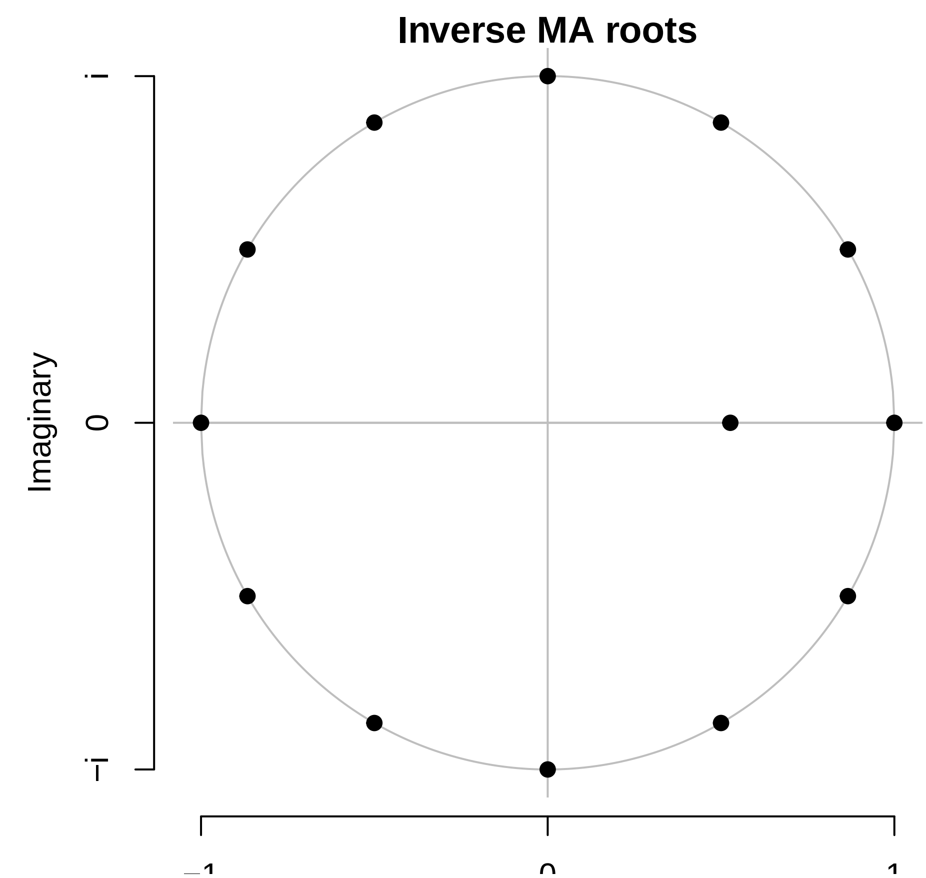



Series resid
1.0
0.5
0.0
−0.5
−1.0
0
1
2
3
4
5


Series resid
1.0
0.5
0.0
−0.5
−1.0
0
1
2
3
4
5




```
##
## --------------------------------------------------------------------
##
## Call:
## arima(x = tserie, order = c(0, 0, 1), seasonal = list(order = c(0, 1, 1), period = 12))
##
## Coefficients:
##           ma1     sma1
##       -0.5268  -1.0000
## s.e.   0.0619   0.0301
##
## sigma^2 estimated as 0.4044:  log likelihood = -308.62,  aic = 623.24
##
## Modul of AR Characteristic polynomial Roots:
##
## Modul of MA Characteristic polynomial Roots:  1 1 1 1 1 1 1 1 1 1 1 1 1.898263
```




```
##
## Psi-weights (MA(inf))
##
## --------------------
##      psi 1      psi 2      psi 3      psi 4      psi 5      psi 6      psi 7
## -0.5267974  0.0000000  0.0000000  0.0000000  0.0000000  0.0000000  0.0000000
##      psi 8      psi 9     psi 10     psi 11     psi 12     psi 13     psi 14
##  0.0000000  0.0000000  0.0000000  0.0000000 -0.9999990  0.5267968  0.0000000
##     psi 15     psi 16     psi 17     psi 18     psi 19     psi 20     psi 21
##  0.0000000  0.0000000  0.0000000  0.0000000  0.0000000  0.0000000  0.0000000
##     psi 22     psi 23     psi 24
##  0.0000000  0.0000000  0.0000000
##
## Pi-weights (AR(inf))
##
## --------------------
##          pi 1          pi 2          pi 3          pi 4          pi 5
## -0.5267973799 -0.2775154795 -0.1461944275 -0.0770148414 -0.0405712166
##          pi 6          pi 7          pi 8          pi 9         pi 10
## -0.0213728106 -0.0112591406 -0.0059312858 -0.0031245858 -0.0016460236
##         pi 11         pi 12         pi 13         pi 14         pi 15
## -0.0008671209 -1.0004557840 -0.5270374858 -0.2776419666 -0.1462610606
##         pi 16         pi 17         pi 18         pi 19         pi 20
## -0.0770499435 -0.0405897084 -0.0213825520 -0.0112642724 -0.0059339892
##         pi 21         pi 22         pi 23         pi 24
## -0.0031260100 -0.0016467739 -0.0008675162 -1.0004549792
```

```
##
## Descriptive Statistics for the Residuals
##
## ----------------------------------------
##                   resid
## nobs        311.000000
## NAs           0.000000
## Minimum      -2.765946
## Maximum       2.867098
## 1. Quartile  -0.390796
## 3. Quartile   0.395781
## Mean         -0.019489
## Median       -0.020791
## Sum          -6.061071
## SE Mean       0.035399
## LCL Mean     -0.089141
## UCL Mean      0.050163
## Variance      0.389705
## Stdev         0.624264
## Skewness      0.140754
## Kurtosis      2.217221
##
## Normality Tests
##
## --------------------
##
##  Shapiro-Wilk normality test
##
## data:  resid
## W = 0.97795, p-value = 0.0001026
##
##
##  Anderson-Darling normality test
##
## data:  resid
## A = 0.91743, p-value = 0.01943
##
##
##  Jarque Bera Test
##
## data:  resid
## X-squared = 66.693, df = 2, p-value = 3.331e-15
##
##
## Homoscedasticity Test
##
## --------------------
##
##  studentized Breusch-Pagan test
##
## data:  resid ~ I(obs - resid)
## BP = 0.037186, df = 1, p-value = 0.8471
##
##
```

```
## Independence Tests
##
## --------------------
##
##  Durbin-Watson test
##
## data:  resid ~ I(1:length(resid))
## DW = 1.9207, p-value = 0.2241
## alternative hypothesis: true autocorrelation is greater than 0
##
##
## Ljung-Box test
##      lag.df   statistic      p.value
## [1,]      1   0.5118169 4.743534e-01
## [2,]      2   0.6025455 7.398759e-01
## [3,]      3   1.3950363 7.066987e-01
## [4,]      4  16.2255699 2.730948e-03
## [5,]     12  95.4284024 4.329870e-15
## [6,]     24 112.8005509 1.821876e-13
## [7,]     36 124.8336412 9.919621e-12
## [8,]     48 144.3940381 1.384759e-11
```

```r
ultim=c(2018,12)
pdq=c(1,1,1)
PDQ=c(0,1,1)

serie2=window(serie,end=ultim)
lnserie2=log(serie2)
serie1=window(serie,end=ultim+c(1,0))
lnserie1=log(serie1)

(modA=arima(lnserie1,order=pdq,seasonal=list(order=PDQ,period=12)))
```

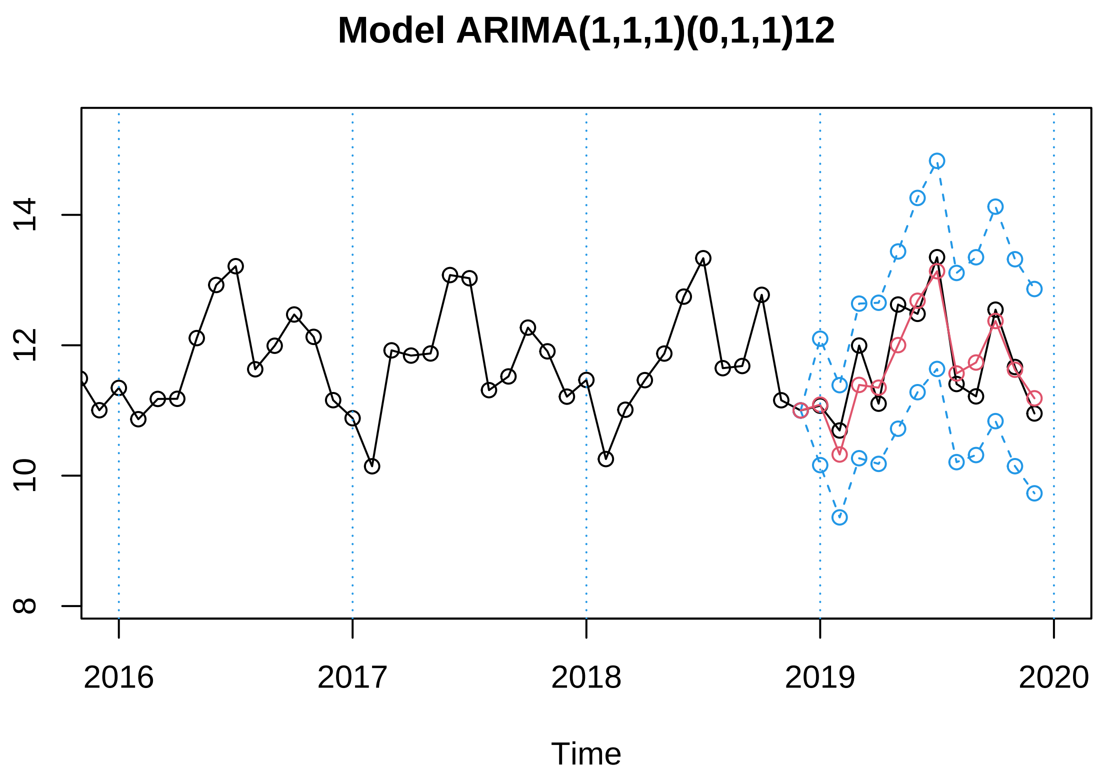
```
##
## Call:
## arima(x = lnserie1, order = pdq, seasonal = list(order = PDQ, period = 12))
##
## Coefficients:
##          ar1      ma1     sma1
##       0.2319  -0.7304  -0.7237
## s.e.  0.0973   0.0706   0.0378
##
## sigma^2 estimated as 0.001953:  log likelihood = 524.1,  aic = -1040.2
```

```r
(modB=arima(lnserie2,order=pdq,seasonal=list(order=PDQ,period=12)))
```

```
##
## Call:
## arima(x = lnserie2, order = pdq, seasonal = list(order = PDQ, period = 12))
##
## Coefficients:
##          ar1      ma1     sma1
##       0.2439  -0.7319  -0.7190
```

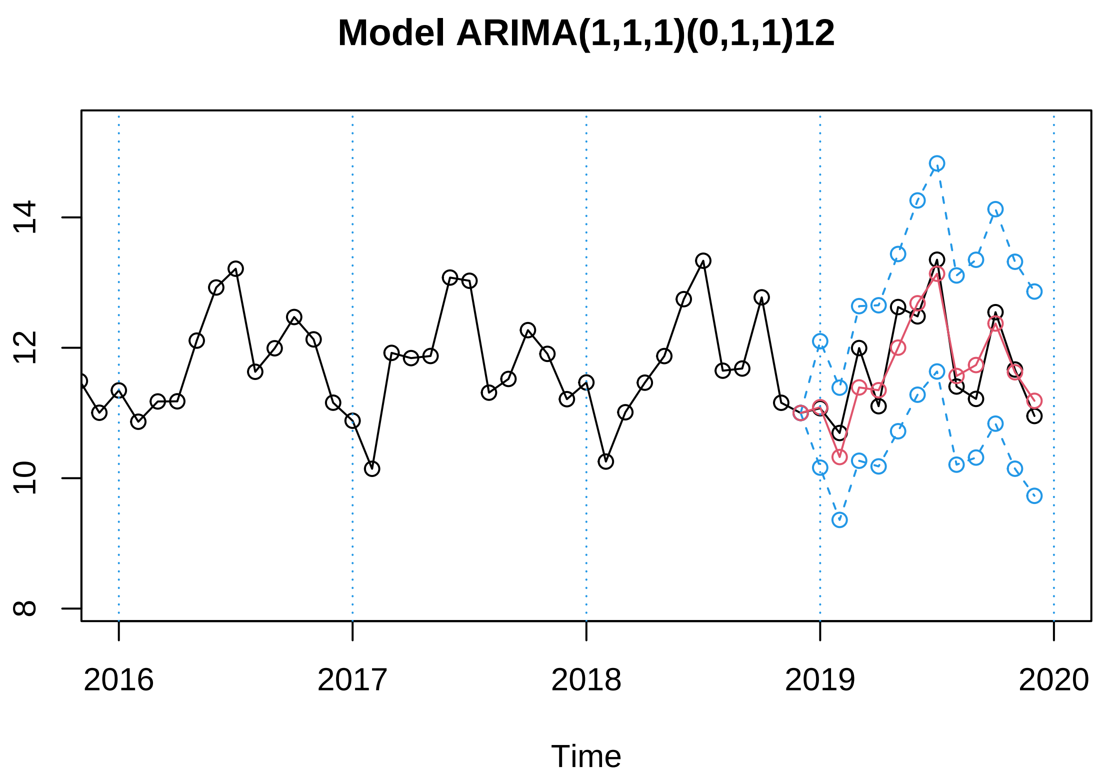
```
## s.e.   0.0998   0.0724   0.0393
##
## sigma^2 estimated as 0.001984:  log likelihood = 501.43,  aic = -994.86
```

```
pred=predict(modB,n.ahead=12)
pr<-ts(c(tail(lnserie2,1),pred$pred),start=ultim,freq=12)
se<-ts(c(0,pred$se),start=ultim,freq=12)

#Intervals
tl<-ts(exp(pr-1.96*se),start=ultim,freq=12)
tu<-ts(exp(pr+1.96*se),start=ultim,freq=12)
pr<-ts(exp(pr),start=ultim,freq=12)


ts.plot(serie,tl,tu,pr,lty=c(1,2,2,1),col=c(1,4,4,2),xlim=ultim[1]+c(-2,+2),type="o",main=paste("Model A
abline(v=(ultim[1]-2):(ultim[1]+2),lty=3,col=4)
```




```
pred=predict(modB,n.ahead=12)
pr<-ts(c(tail(lnserie2,1),pred$pred),start=ultim,freq=12)
se<-ts(c(0,pred$se),start=ultim,freq=12)

#Intervals
tl<-ts(exp(pr-1.96*se),start=ultim,freq=12)
tu<-ts(exp(pr+1.96*se),start=ultim,freq=12)
pr<-ts(exp(pr),start=ultim,freq=12)
```

```
ts.plot(serie,tl,tu,pr,lty=c(1,2,2,1),col=c(1,4,4,2),xlim=ultim[1]+c(-2,+2),type="o",main=paste("Model A
abline(v=(ultim[1]-2):(ultim[1]+2),lty=3,col=4)
```




```
obs=window(serie,start=ultim+c(0,1))
pr=window(pr,start=ultim+c(0,1))
ts(data.frame(LowLim=tl[-1],Predic=pr,UpperLim=tu[-1],Observ=obs,Error=obs-pr,PercentError=(obs-pr)/obs)
```

```
##             LowLim   Predic UpperLim     V1         obs  X.obs...pr.
## Jan 2019 10.162032 11.08905 12.10062 11.073 -0.01604597 -0.001449108
## Feb 2019  9.360074 10.32458 11.38848 10.693  0.36841777  0.034454107
## Mar 2019 10.267649 11.39185 12.63915 11.996  0.60414711  0.050362380
## Apr 2019 10.180903 11.34937 12.65195 11.104 -0.24537380 -0.022097784
## May 2019 10.719810 12.00256 13.43881 12.625  0.62244055  0.049302222
## Jun 2019 11.280301 12.68295 14.26002 12.482 -0.20095239 -0.016099374
## Jul 2019 11.636391 13.13604 14.82896 13.351  0.21496092  0.016100735
## Aug 2019 10.207447 11.56785 13.10956 11.406 -0.16184978 -0.014189881
## Sep 2019 10.316305 11.73540 13.34970 11.216 -0.51939885 -0.046308742
## Oct 2019 10.836546 12.37246 14.12607 12.546  0.17353953  0.013832260
## Nov 2019 10.145426 11.62476 13.31981 11.667  0.04223557  0.003620088
## Dec 2019  9.729313 11.18678 12.86258 10.954 -0.23278177 -0.021250846
```

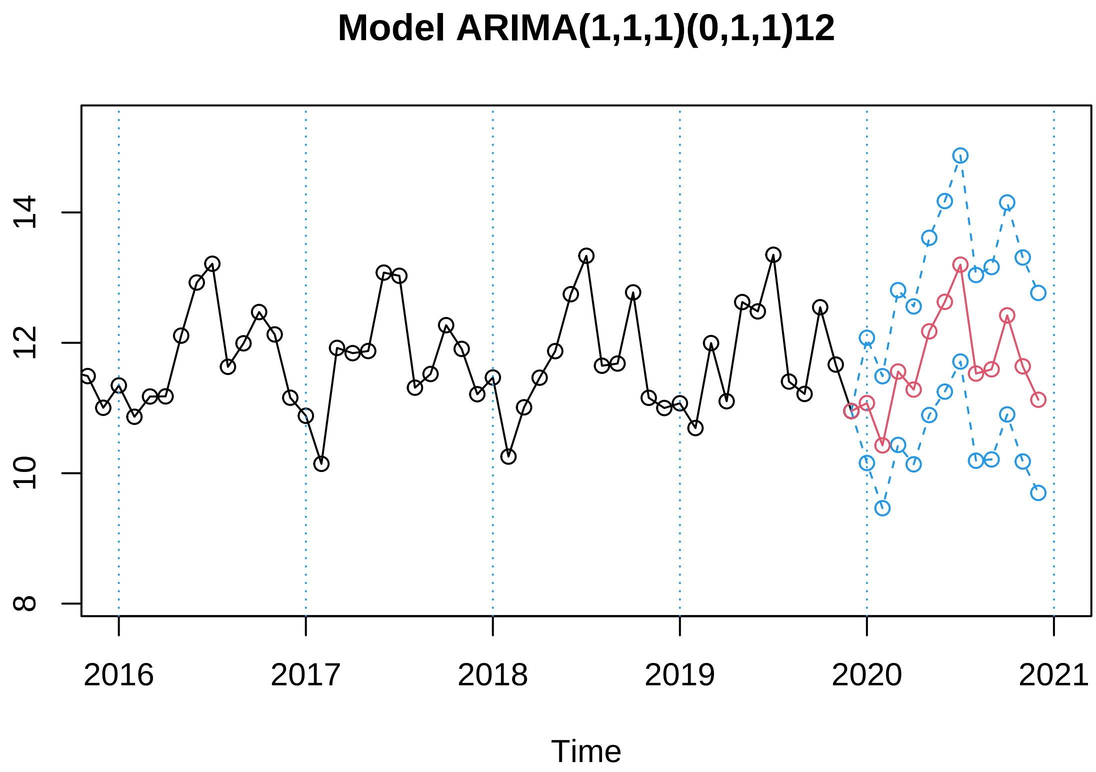
```r
mod.RMSE1=sqrt(sum((obs-pr)^2)/12)
mod.MAE1=sum(abs(obs-pr))/12
mod.RMSPE1=sqrt(sum(((obs-pr)/obs)^2)/12)
mod.MAPE1=sum(abs(obs-pr)/obs)/12

data.frame("RMSE"=mod.RMSE1,"MAE"=mod.MAE1,"RMSPE"=mod.RMSPE1,"MAPE"=mod.MAPE1)
```

```
##        RMSE      MAE      RMSPE       MAPE
## 1 0.3436654 0.283512 0.02910453 0.02408896
```

```r
mCI1=mean(tu-tl)

cat("\nMean Length CI: ",mCI1)
```

```
##
## Mean Length CI:  2.556423
```

```r
pred=predict(modA,n.ahead=12)
pr<-ts(c(tail(lnserie1,1),pred$pred),start=ultim+c(1,0),freq=12)
se<-ts(c(0,pred$se),start=ultim+c(1,0),freq=12)

tl1<-ts(exp(pr-1.96*se),start=ultim+c(1,0),freq=12)
tu1<-ts(exp(pr+1.96*se),start=ultim+c(1,0),freq=12)
pr1<-ts(exp(pr),start=ultim+c(1,0),freq=12)

ts.plot(serie,tl1,tu1,pr1,lty=c(1,2,2,1),col=c(1,4,4,2),xlim=c(ultim[1]-2,ultim[1]+3),type="o",main=past
abline(v=(ultim[1]-2):(ultim[1]+3),lty=3,col=4)
```




```
(previs1=window(cbind(tl1,pr1,tu1),start=ultim+c(1,0)))
```

```
##                tl1      pr1      tu1
## Dec 2019 10.954000 10.95400 10.95400
## Jan 2020 10.156638 11.07557 12.07764
## Feb 2020  9.464124 10.42706 11.48798
## Mar 2020 10.433898 11.56010 12.80787
## Apr 2020 10.136856 11.28296 12.55864
## May 2020 10.892537 12.17603 13.61075
## Jun 2020 11.251434 12.62869 14.17454
## Jul 2020 11.713183 13.19883 14.87292
## Aug 2020 10.191999 11.52855 13.04037
## Sep 2020 10.211567 11.59345 13.16233
## Oct 2020 10.901549 12.42133 14.15297
## Nov 2020 10.180144 11.63996 13.30910
## Dec 2020  9.698125 11.12665 12.76559
```

```
ultim=c(2018,12)
pdq=c(8,1,0)
PDQ=c(0,1,1)

serie2=window(serie,end=ultim)
lnserie2=log(serie2)
serie1=window(serie,end=ultim+c(1,0))
lnserie1=log(serie1)
```

```r
(modA=arima(lnserie1,order=pdq,seasonal=list(order=PDQ,period=12)))
```

```
##
## Call:
## arima(x = lnserie1, order = pdq, seasonal = list(order = PDQ, period = 12))
##
## Coefficients:
##           ar1      ar2      ar3      ar4      ar5      ar6      ar7      ar8
##       -0.5201  -0.3314  -0.1937  -0.2224  -0.1850  -0.2485  -0.1674  -0.0184
## s.e.   0.0570   0.0637   0.0646   0.0650   0.0642   0.0650   0.0637   0.0573
##          sma1
##       -0.7274
## s.e.   0.0392
##
## sigma^2 estimated as 0.001874:  log likelihood = 530.32,  aic = -1040.64
```

```r
(modB=arima(lnserie2,order=pdq,seasonal=list(order=PDQ,period=12)))
```

```
##
## Call:
## arima(x = lnserie2, order = pdq, seasonal = list(order = PDQ, period = 12))
##
## Coefficients:
##           ar1      ar2      ar3      ar4      ar5      ar6      ar7      ar8
##       -0.5103  -0.3269  -0.1857  -0.2154  -0.1920  -0.2358  -0.1678  -0.0145
## s.e.   0.0581   0.0649   0.0659   0.0661   0.0653   0.0662   0.0647   0.0584
##          sma1
##       -0.7225
## s.e.   0.0408
##
## sigma^2 estimated as 0.001908:  log likelihood = 507.09,  aic = -994.17
```