Covid 19 Mortalità

Boschi Giulia 804623

1/5/2020

```
library(dplyr)
library(tidyr)
library(ggplot2)
suppressPackageStartupMessages(library(ggplot2))
suppressPackageStartupMessages(library(forecast))
suppressPackageStartupMessages(library(astsa))
suppressPackageStartupMessages(library(lmtest))
suppressPackageStartupMessages(library(fUnitRoots))
suppressPackageStartupMessages(library(FitARMA))
suppressPackageStartupMessages(library(strucchange))
suppressPackageStartupMessages(library(reshape))
suppressPackageStartupMessages(library(Rmisc))
suppressPackageStartupMessages(library(fBasics))
library(tseries)
library(lubridate)

setwd("C:\\Users\\giuli\\Google Drive\\DS\\Data science lab\\dati-giornalieri-
comune")

dt <- read.csv("comune_giorno.csv")

td <- dt %>% gather(key = "SESSO_ANNO", value = "DECESSI", #diamo dei nomi
                    MASCHI_15:TOTALE_20)

splt_sesso_anno <- strsplit(td$SESSO_ANNO, "_", fixed = T)

td <- td %>% mutate(SESSO = sapply(splt_sesso_anno, function(x) x[1]),
                     ANNO = sapply(splt_sesso_anno, function(x) x[2])) %>%
  select(-SESSO_ANNO)

td %>% mutate(DATA = as.Date(paste0("0", GE, "20", ANNO), format = "%m%d%Y")) ->
td
```

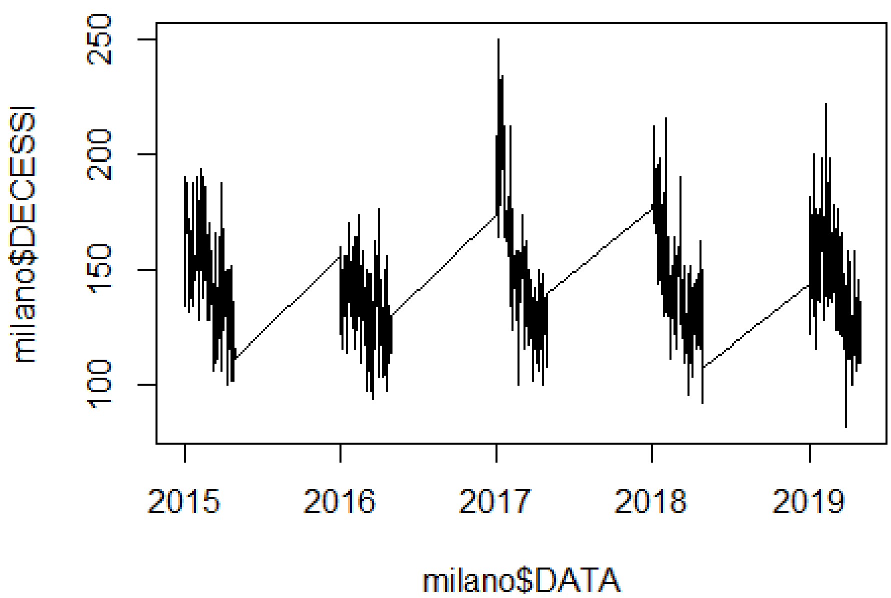
#import td

```
td <- read.csv("td.csv")

td %>% dplyr::filter(DATA_INIZIO_DIFF == "1 aprile",
              NOME_REGIONE == "Lombardia",
              DECESSI < 9999) -> wdt

wdt %>% dplyr::group_by(DATA, ANNO, NOME_PROVINCIA) %>%
  dplyr::summarise(DECESSI = sum(DECESSI)) %>%
  dplyr::arrange(ANNO, DATA)-> wdt
```

```r
write.csv(wdt, "C:\\Users\\giuli\\Google Drive\\DS\\Data science lab\\dati-giornalieri-comune\\Lombardia.csv", row.names = F)
```

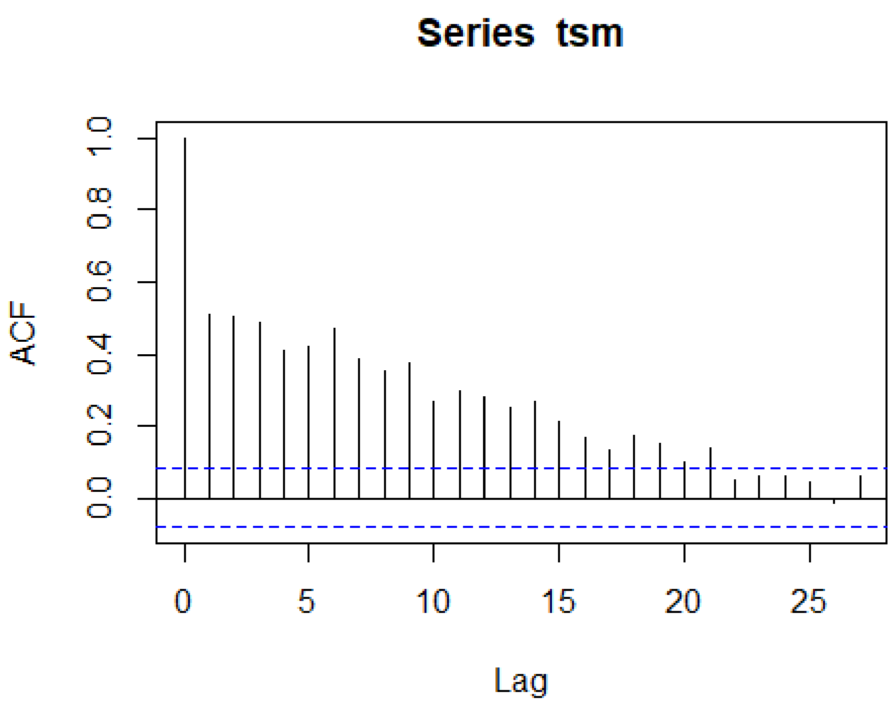
SERIE STORICA MILANO fino al 2019

```r
Lombardia <- read.csv("Lombardia.csv", colClasses = c('Date', 'integer', 'character', 'integer'))

milano <- Lombardia %>% dplyr::filter(NOME_PROVINCIA == "Milano",  DATA < "2020-01-01")

plot(milano$DATA, milano$DECESSI, type = "l")
```

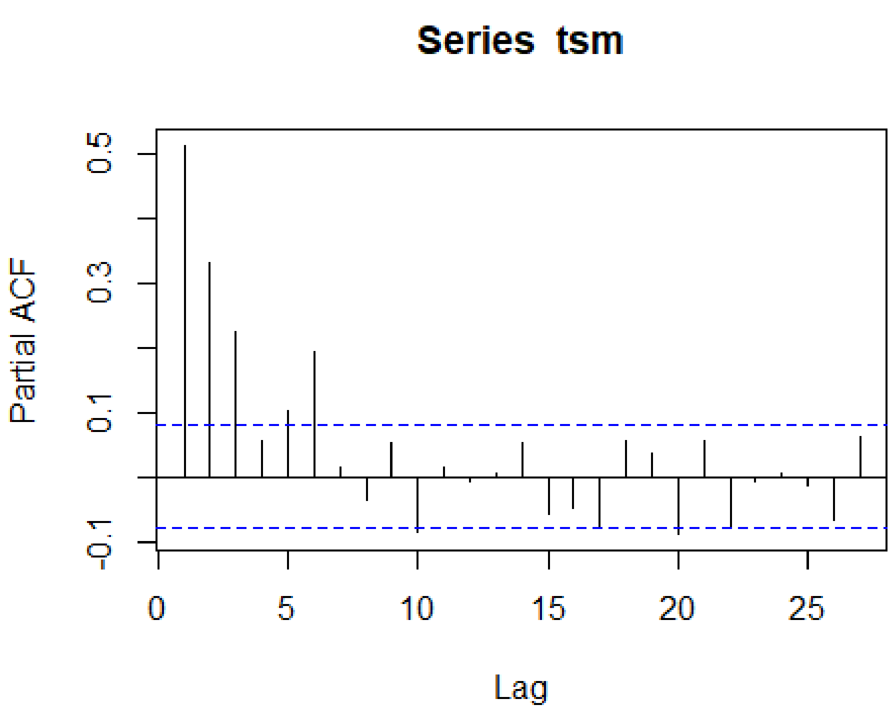


Andamento non stazionario in varianza (valutare una trasformazione logaritmica). Gli ultimi tre anni hanno varianza maggiore rispetto ai primi 2, si osserva che il 2016 è praticamente stazionario se non fosse per la fine del periodo che cala leggermente. 2017 hanno von carianza maggiore (valutiamo se inserire intervento). Escludendo il 2017, il trend non mi sembra particolarmente crescente.

```r
tsm <- ts(milano$DECESSI)

acf(tsm)
```




```
pacf(tsm)
```



Acf si annulla lentamente (sintomi di non stazionarietà), pacf si annulla al terzo lag ma esce per due lag successivi (forse c'è una regolarità). In acf si osserva l'andamento stagionale (onde regolari sui lag, ma di difficile interpretazione data la particolare serie). Da questi correllogrammi direi che c'è della stagionalità che andrebbe corretta, ma non me la sento di fare differenze dato che si tratta di 5 serie separate temporalmente una dall'altra. L'alternativa che si può provare è inserire componenti stagionali SAR e SMA nel modello SARIMA.

```
adf.test(tsm)

##
##  Augmented Dickey-Fuller Test
##
## data:  tsm
## Dickey-Fuller = -3.7864, Lag order = 8, p-value = 0.01968
## alternative hypothesis: stationary
```

Test di Dickey fuller per capire se la serie presenta una radice unitaria ($|| < 0 $) che comporta la non stazionarietà del modello. In questo caso siamo sul limite. L'obiettivo sarebbe avere un p-val < 0.01, il nostro è 0.02.

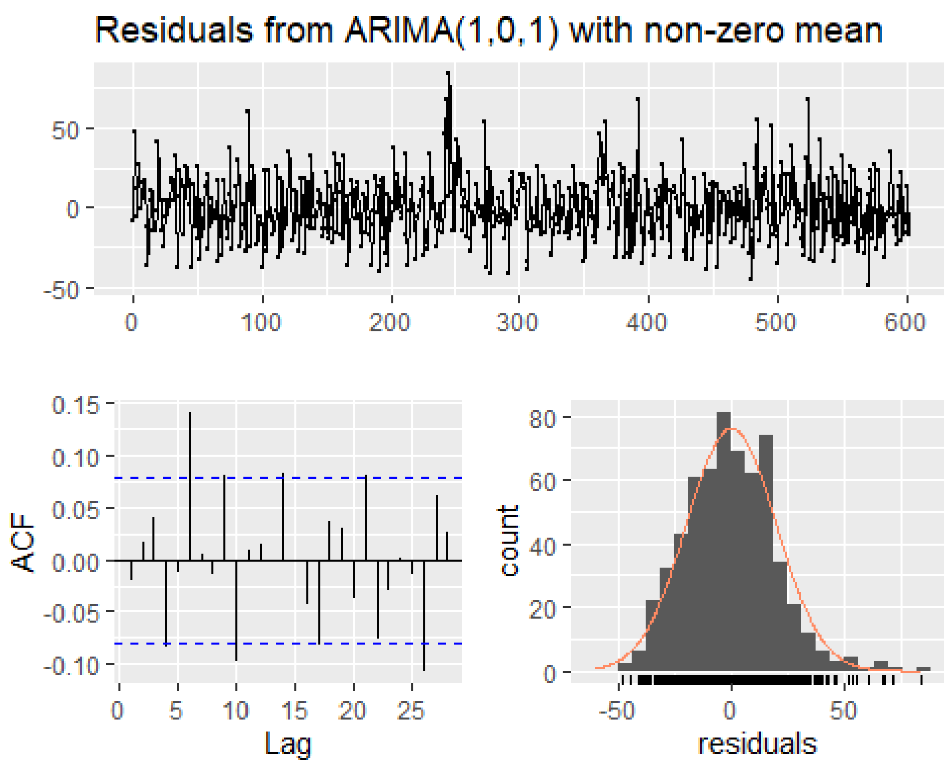
#Modello semplicissimo

```
mod1 <- Arima(tsm, order = c(1, 0, 1))

summary(mod1)

## Series: tsm
## ARIMA(1,0,1) with non-zero mean
##
## Coefficients:
##          ar1      ma1      mean
##       0.9521  -0.7051  144.1047
## s.e.  0.0161   0.0352    4.8393
##
## sigma^2 estimated as 394.2:  log likelihood=-2647.75
## AIC=5303.51   AICc=5303.58   BIC=5321.1
##
## Training set error measures:
##                       ME     RMSE      MAE       MPE    MAPE      MASE
## Training set -0.03580875 19.80532 15.58597 -1.891143 11.0579 0.7837398
##                     ACF1
## Training set -0.02103004

coeftest(mod1)

##
## z test of coefficients:
##
##             Estimate Std. Error z value  Pr(>|z|)
## ar1         0.952054   0.016076  59.224 < 2.2e-16 ***
## ma1        -0.705076   0.035244 -20.006 < 2.2e-16 ***
## intercept 144.104705   4.839337  29.778 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Parametri significativi, AIC = 5303, MAPE = 11 (Indicatore di bontà del modello, di solito si ritiene buono un modello con MAPE inferiore a 12 o 15)

Analisi dei residui

```r
checkresiduals(mod1)
```




```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(1,0,1) with non-zero mean
## Q* = 27.913, df = 7, p-value = 0.000228
##
## Model df: 3.   Total lags used: 10
```

```r
#test incorrelazione
LjungBoxTest(residuals(mod1), k = 1, lag.max = 20)
```

```
##   m    Qm       pvalue
##   1  0.27 0.6052643511
##   2  0.45 0.5005534971
##   3  1.46 0.4810679799
##   4  5.69 0.1274917669
##   5  5.78 0.2159807007
##   6 17.86 0.0031245733
##   7 17.88 0.0065268443
##   8 18.02 0.0118643530
##   9 22.00 0.0049090542
##  10 27.91 0.0009862984
```

```
##  11 27.97 0.0018226121
##  12 28.11 0.0031088292
##  13 28.12 0.0053124619
##  14 32.45 0.0020599501
##  15 32.45 0.0034563706
##  16 33.62 0.0038515010
##  17 37.90 0.0015658733
##  18 38.69 0.0019755683
##  19 39.29 0.0026063392
##  20 40.20 0.0030833591
```

```
#test normalità
jarque.bera.test(residuals(mod1))
```

```
##
##  Jarque Bera Test
##
## data:  residuals(mod1)
## X-squared = 35.653, df = 2, p-value = 1.811e-08
```

Test di Ljung box testa $H_0: \rho_1 = \rho_2 = \ldots = \rho_n = 0$ per vedere se i residui sono incorrelati tra loro e quindi realizzazione di un processo white noise. Il test rifiuta l'ipotesi nulla quindi sono correlati però, guardando i grafici, sembrerebbero white noise (il plot salta su e giù intorno allo 0 e acf è nullo ad eccezione di qualche lag). Quello che fa saltare il test è proprio l'acf che è significativo per qualche lag con una certa "stagionalità". Ci consiglia di correggerla, ma non sa che non è una vera stagionalità.

Dal test di Jarque-Bera per la normalità (H_0: i residui sono realizzazione di una v.c. normale) si deduce che i residui non sono normali (p-val < 0.01). Però, osservando il grafico dei residui e il loro adattamento alla curva della normale, si nota un ottimo adattamento a quest'ultima. Il problema che rileva il test è probabilmente causato dalle code. Non mi preoccuperei della non normalità visto l'adattamento.

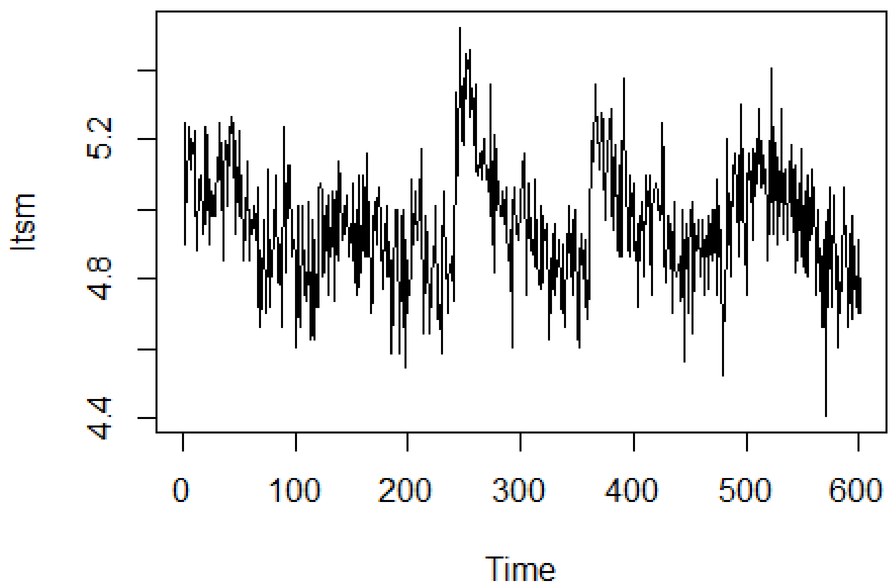
#Modello con trasformata logaritmica

```
BoxCox.lambda(tsm)
```

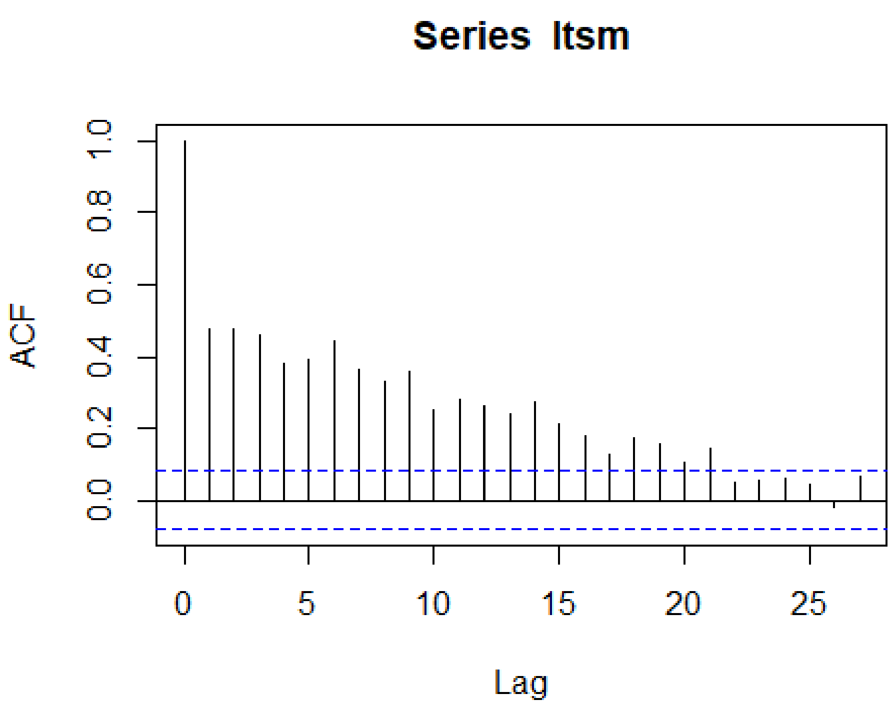
```
## [1] 0.2229013
```

Il lambda ottimo è abbastanza lontano dallo o (trasformazione logaritmica), è praticamente a metà strada da 0.5

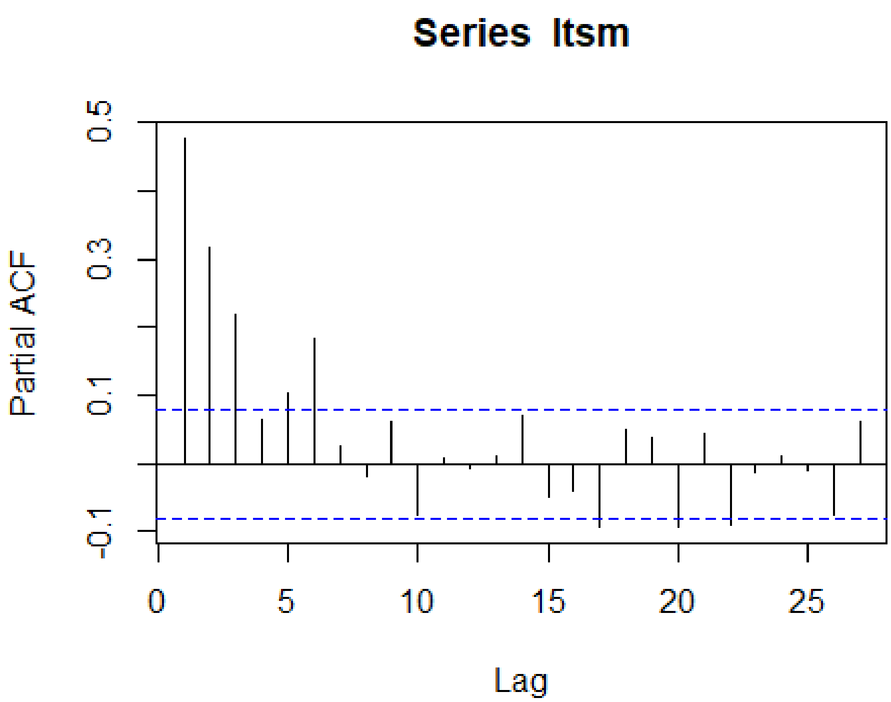
```
ltsm <- log(tsm)

ts.plot(ltsm)
```




```
acf(ltsm)
```




```
pacf(ltsm)
```




```
adf.test(ltsm)

##
##  Augmented Dickey-Fuller Test
##
## data:  ltsm
## Dickey-Fuller = -3.7621, Lag order = 8, p-value = 0.02089
## alternative hypothesis: stationary
```

Si vedono ancora i due salti sul 2017 e 2018, ma il problema è sempre che non è corretto vederla così I correlogrammi hanno lo stesso identico andamento di quelli senza trasformata logaritmica e il p-value del test di DF aumenta (non drasticamente, ma se non andava bene prima non va bene neanche ora).

```
mod2 <- Arima(ltsm, order = c(1, 0, 1))

summary(mod2)

## Series: ltsm
## ARIMA(1,0,1) with non-zero mean
##
## Coefficients:
##          ar1      ma1    mean
##       0.9534  -0.7248  4.9575
## s.e.  0.0162   0.0351  0.0321
##
## sigma^2 estimated as 0.01887:  log likelihood=341.3
## AIC=-674.6   AICc=-674.54   BIC=-657.01
```

```
##
## Training set error measures:
##                         ME      RMSE       MAE         MPE     MAPE      MASE
## Training set -0.0006337499 0.1370392 0.1091165 -0.08955508 2.207454 0.7831546
##                     ACF1
## Training set -0.01990818
```

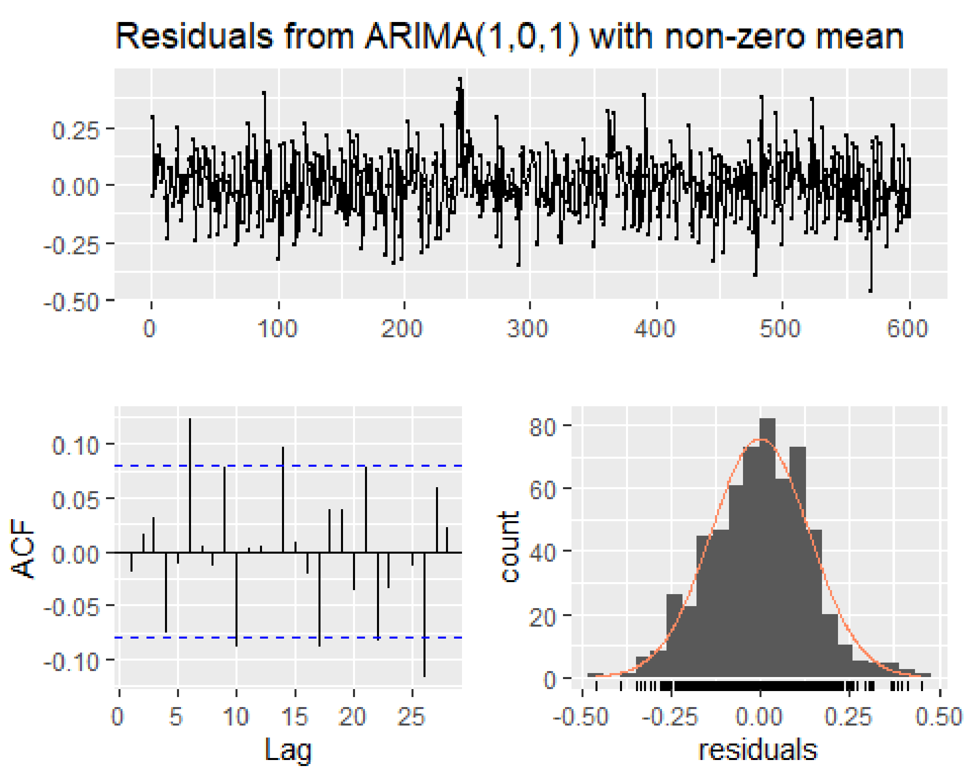
```
coeftest(mod2)
```

```
##
## z test of coefficients:
##
##            Estimate Std. Error z value  Pr(>|z|)
## ar1         0.953388   0.016207  58.826 < 2.2e-16 ***
## ma1        -0.724778   0.035101 -20.648 < 2.2e-16 ***
## intercept  4.957540   0.032084 154.516 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Nonostante i presupposti per la trasformazione logaritmica non fossero ottimi il valore di AIC è crollato (-674) e questo è ottimo e anche il valore di MAPE si è ridotto sensibilmente (2.2). I coefficienti significativi sono gli stessi del modello 1.

Analisi dei residui

```
checkresiduals(mod2)
```




```
##
##  Ljung-Box test
```

```
##
## data:  Residuals from ARIMA(1,0,1) with non-zero mean
## Q* = 22.484, df = 7, p-value = 0.002095
##
## Model df: 3.   Total lags used: 10
```

```r
#test incorrelazione
LjungBoxTest(residuals(mod2), k = 1, lag.max = 20)
```

```
##   m    Qm      pvalue
##   1  0.24 0.624647968
##   2  0.41 0.519922053
##   3  1.02 0.600064407
##   4  4.51 0.211611624
##   5  4.60 0.331243747
##   6 13.82 0.016806054
##   7 13.83 0.031555226
##   8 13.96 0.051899708
##   9 17.74 0.023233643
##  10 22.48 0.007464903
##  11 22.49 0.012795519
##  12 22.51 0.020700140
##  13 22.51 0.032138026
##  14 28.32 0.008165325
##  15 28.36 0.012723425
##  16 28.64 0.017884362
##  17 33.45 0.006439479
##  18 34.35 0.007562609
##  19 35.33 0.008590313
##  20 36.17 0.010048222
```

```r
#test normalità
jarque.bera.test(residuals(mod2))
```

```
##
##  Jarque Bera Test
##
## data:  residuals(mod2)
## X-squared = 0.85194, df = 2, p-value = 0.6531
```

Il p-value di LB è cresciuto (Q diminuisce di 5 punti), perà i residui restano correlati, ma ora sono normali (JB test ha p-val = 0.65)

#Modello con trasformata logaritmica ed inserimento di stagionalità SARIMA Questa è proprio farla sporca. Per fare il modello sarima, devi mettere nella lista la stagionalità. Io ho messo 4 per indicare che si tratta di trimesti, lui però così li interpreta come trimestri dello stesso anno

```r
mod3 <- Arima(ltsm, order = c(1, 0, 1), seasonal = list(order = c(1,0,0), period =
4))

summary(mod3)
```

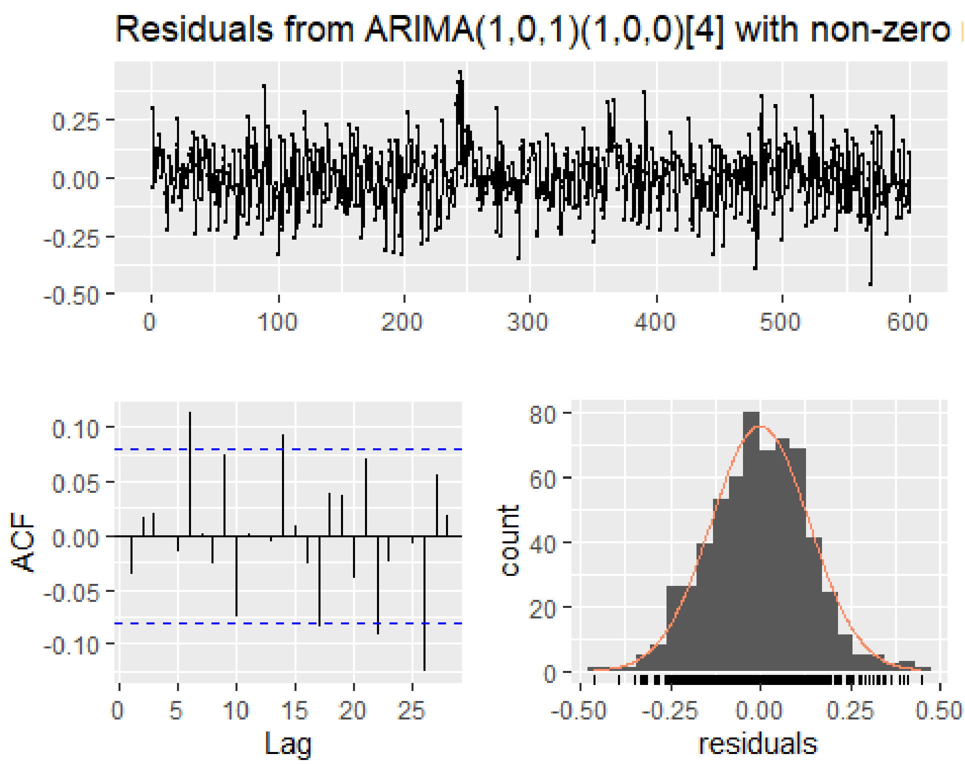
```
## Series: ltsm
## ARIMA(1,0,1)(1,0,0)[4] with non-zero mean
##
## Coefficients:
##          ar1      ma1     sar1    mean
##       0.9568  -0.7128  -0.0861  4.9581
## s.e.  0.0150   0.0336   0.0426  0.0331
##
## sigma^2 estimated as 0.01878:  log likelihood=343.31
## AIC=-676.62   AICc=-676.52   BIC=-654.62
##
## Training set error measures:
##                         ME     RMSE       MAE         MPE     MAPE      MASE
## Training set -0.0008006793 0.136579 0.1087153 -0.09237605 2.199343 0.7802748
##                     ACF1
## Training set -0.0354612
```

```r
coeftest(mod3)
```

```
##
## z test of coefficients:
##
##            Estimate Std. Error  z value Pr(>|z|)
## ar1        0.956810   0.014972  63.9085  < 2e-16 ***
## ma1       -0.712755   0.033641 -21.1869  < 2e-16 ***
## sar1      -0.086063   0.042607  -2.0199  0.04339 *
## intercept  4.958106   0.033076 149.9019  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Il coefficiente autoregressivo per la componente stagionale è poco significativo. AIC scende ancora, ma di poco, rispetto al modello prima e MAPE non ha un netto miglioramento.

```r
checkresiduals(mod3)
```




```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(1,0,1)(1,0,0)[4] with non-zero mean
## Q* = 16.447, df = 6, p-value = 0.01155
##
## Model df: 4.   Total lags used: 10
```

```
#test incorrelazione
LjungBoxTest(residuals(mod3), k = 1, lag.max = 20)
```

```
##    m    Qm     pvalue
##    1  0.76 0.38347426
##    2  0.92 0.33641452
##    3  1.18 0.55486967
##    4  1.18 0.75815519
##    5  1.32 0.85835523
##    6  9.17 0.10244081
##    7  9.17 0.16402973
##    8  9.59 0.21301664
##    9 13.04 0.11052907
##   10 16.45 0.05812074
##   11 16.45 0.08752924
##   12 16.45 0.12532742
##   13 16.47 0.17056706
##   14 21.77 0.05896263
##   15 21.84 0.08205140
##   16 22.25 0.10148076
```

```
##  17 26.77 0.04405285
##  18 27.76 0.04786114
##  19 28.64 0.05291293
##  20 29.67 0.05620735
```

```r
#test normalità
jarque.bera.test(residuals(mod3))
```

```
##
##  Jarque Bera Test
##
## data:  residuals(mod3)
## X-squared = 0.87017, df = 2, p-value = 0.6472
```

Ci avviciniamo all'incorrelazione e sono normali. Non inserirei la stagionalità per dei così piccoli miglioramenti dato che siamo in questa situazione strana e comunque non corregge l'incorrelazione (unico vero problema che abbiamo).

#Modello log con differenza prima Per tentare di correggere la correlazione inserisco una differenza prima

```r
mod4 <- Arima(ltsm, order = c(0, 1, 1))

summary(mod4)
```

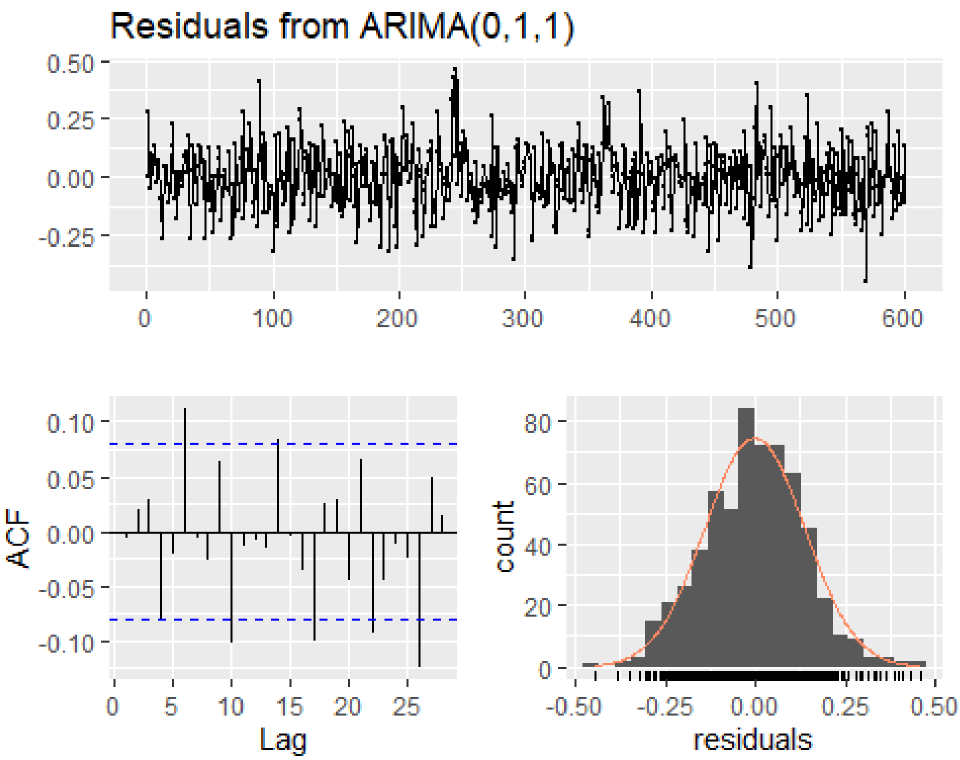
```
## Series: ltsm
## ARIMA(0,1,1)
##
## Coefficients:
##           ma1
##       -0.7642
## s.e.   0.0267
##
## sigma^2 estimated as 0.01919:  log likelihood=334.69
## AIC=-665.38   AICc=-665.36   BIC=-656.59
##
## Training set error measures:
##                        ME      RMSE      MAE        MPE     MAPE      MASE
## Training set -0.001622034 0.1383013 0.109784 -0.1020422 2.220529 0.7879451
##                      ACF1
## Training set -0.006956749
```

```r
coeftest(mod4)
```

```
##
## z test of coefficients:
##
##      Estimate Std. Error z value  Pr(>|z|)
## ma1 -0.764152   0.026683 -28.639 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Inserendo la differenza prima l'unico coefficiente che resta significativo è quello a media mobile. I valori di MAPE e AIC non mostrano chissà che miglioramenti

```r
checkresiduals(mod4)
```




```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(0,1,1)
## Q* = 21.878, df = 9, p-value = 0.009273
##
## Model df: 1.   Total lags used: 10
```

```r
#test incorrelazione
LjungBoxTest(residuals(mod4), k = 1, lag.max = 20)
```

```
##   m    Qm      pvalue
##   1  0.03 0.864245288
##   2  0.25 0.615278204
##   3  0.73 0.692806595
##   4  4.77 0.189208981
##   5  5.02 0.285370411
##   6 12.59 0.027541554
##   7 12.61 0.049635819
##   8 13.03 0.071460626
##   9 15.58 0.048818900
##  10 21.88 0.009272756
##  11 21.98 0.015214593
##  12 22.02 0.024191310
```

```
##  13 22.15 0.035821518
##  14 26.52 0.014477182
##  15 26.53 0.022171101
##  16 27.28 0.026521253
##  17 33.62 0.006119094
##  18 34.00 0.008392132
##  19 34.53 0.010823851
##  20 35.83 0.011071045
```

```
#test normalità
jarque.bera.test(residuals(mod4))
```

```
##
##  Jarque Bera Test
##
## data:  residuals(mod4)
## X-squared = 0.96827, df = 2, p-value = 0.6162
```

Residui sempre normali, ma correlati

#Modello con differenza e stagionalità

```
mod5 <- Arima(ltsm, order = c(0, 1, 1), seasonal = list(order = c(1,0,1), period =
4))

summary(mod5)
```

```
## Series: ltsm
## ARIMA(0,1,1)(1,0,1)[4]
##
## Coefficients:
##           ma1    sar1     sma1
##       -0.7472  0.7178  -0.7985
## s.e.   0.0279  0.1704   0.1467
##
## sigma^2 estimated as 0.01901:  log likelihood=338.43
## AIC=-668.85   AICc=-668.78   BIC=-651.26
##
## Training set error measures:
##                        ME     RMSE       MAE        MPE     MAPE      MASE
## Training set -0.002112167 0.137426 0.1089113 -0.1130983 2.203215 0.7816815
##                     ACF1
## Training set -0.02509501
```

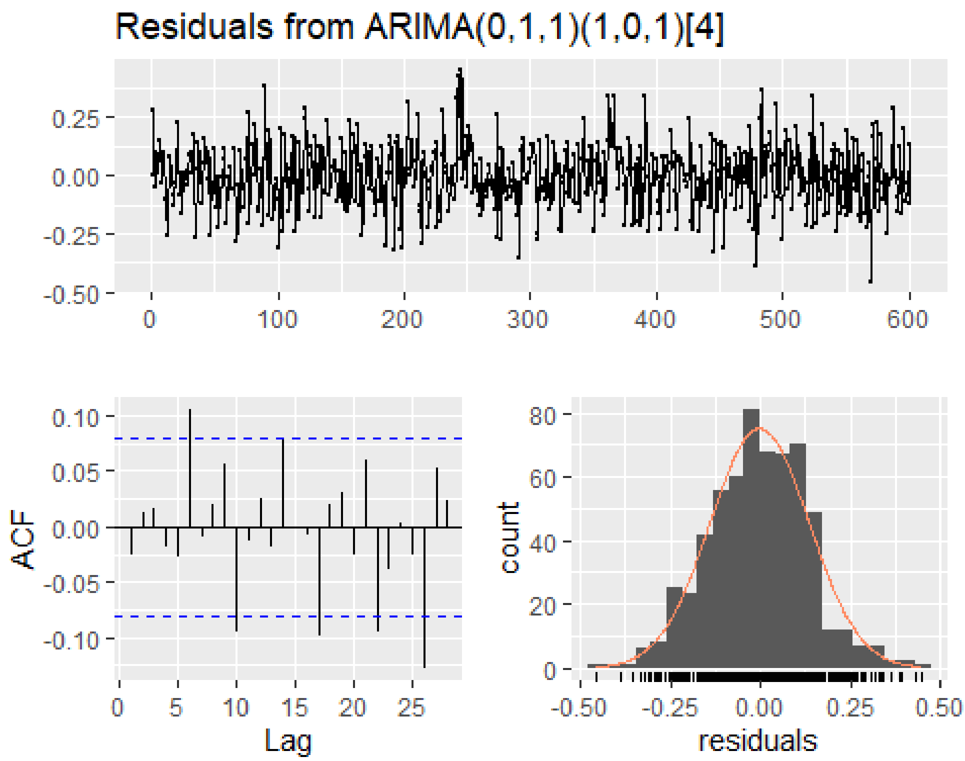
```
coeftest(mod5)
```

```
##
## z test of coefficients:
##
##       Estimate Std. Error  z value  Pr(>|z|)
## ma1  -0.747167   0.027904 -26.7766 < 2.2e-16 ***
## sar1  0.717777   0.170367   4.2131 2.519e-05 ***
## sma1 -0.798457   0.146666  -5.4440 5.208e-08 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

MAPE e AIC sono sempre lì

```
checkresiduals(mod5)
```




```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(0,1,1)(1,0,1)[4]
## Q* = 15.752, df = 7, p-value = 0.02749
##
## Model df: 3.   Total lags used: 10

#test incorrelazione
LjungBoxTest(residuals(mod5), k = 1, lag.max = 20)

##   m    Qm      pvalue
##   1  0.38 0.53740102
##   2  0.47 0.49090536
##   3  0.62 0.73332152
##   4  0.82 0.84441268
##   5  1.27 0.86628378
##   6  8.00 0.15629128
##   7  8.06 0.23364250
##   8  8.31 0.30614939
##   9 10.25 0.24758696
##  10 15.75 0.07225487
```

```
##  11 15.86 0.10359411
##  12 16.28 0.13121602
##  13 16.48 0.17024586
##  14 20.18 0.09073877
##  15 20.19 0.12430870
##  16 20.23 0.16320782
##  17 26.35 0.04924989
##  18 26.59 0.06442627
##  19 27.16 0.07613327
##  20 27.58 0.09190177
```

```
#test normalità
jarque.bera.test(residuals(mod5))
```

```
##
##  Jarque Bera Test
##
## data:  residuals(mod5)
## X-squared = 1.1717, df = 2, p-value = 0.5566
```

Miglioriamo sempre di più il valore di Q, ci avviciniamo alla incorrelazione, anzi potremmo anche accettarla per una soglia un po' scarsa, ma sinceramente non so se sia giusto questo modello sia per la differenza che per la stagionalità fasulla.

#Modello con intervento sul 2017 Aggiungo una colonna al dataset con valore 1 quando siamo nel 2017 e 0 negli altri anni

```
milano$int <- ifelse(year(milano$DATA) == "2017", 1, 0)

mod6 <- Arima(tsm, order = c(1, 0, 1), xreg = milano$int)

summary(mod6)
```

```
## Series: tsm
## Regression with ARIMA(1,0,1) errors
##
## Coefficients:
##          ar1      ma1  intercept    xreg
##       0.9507  -0.7048   142.7535  8.4504
## s.e.  0.0164   0.0357     4.9355  7.8252
##
## sigma^2 estimated as 394.1:  log likelihood=-2647.17
## AIC=5304.34   AICc=5304.44   BIC=5326.33
##
## Training set error measures:
##                       ME     RMSE      MAE       MPE     MAPE      MASE
## Training set -0.08508959 19.78639 15.61259 -1.925189 11.07946 0.7850781
##                     ACF1
## Training set -0.02188911
```

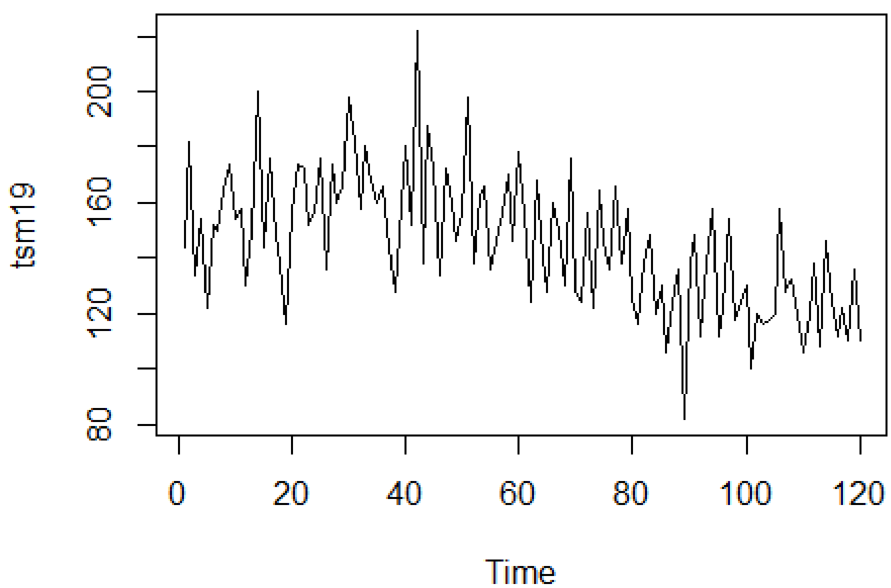
```
coeftest(mod6)
```

```
##
## z test of coefficients:
##
##              Estimate Std. Error  z value Pr(>|z|)
## ar1          0.950741   0.016377  58.0535   <2e-16 ***
## ma1         -0.704842   0.035651 -19.7705   <2e-16 ***
## intercept 142.753463   4.935504  28.9238   <2e-16 ***
## xreg         8.450375   7.825196   1.0799   0.2802
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

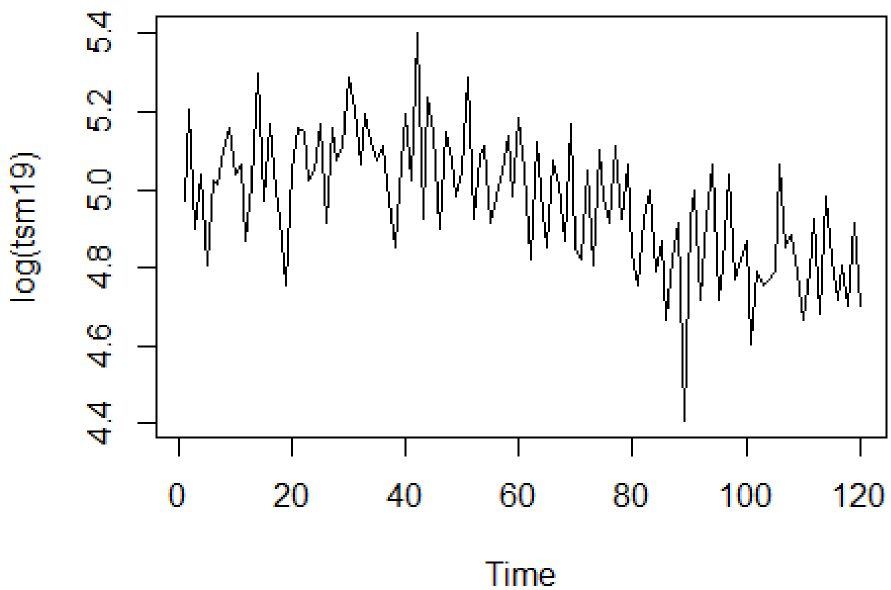
L'intervento non è significativo

#Modello solo per il 2019

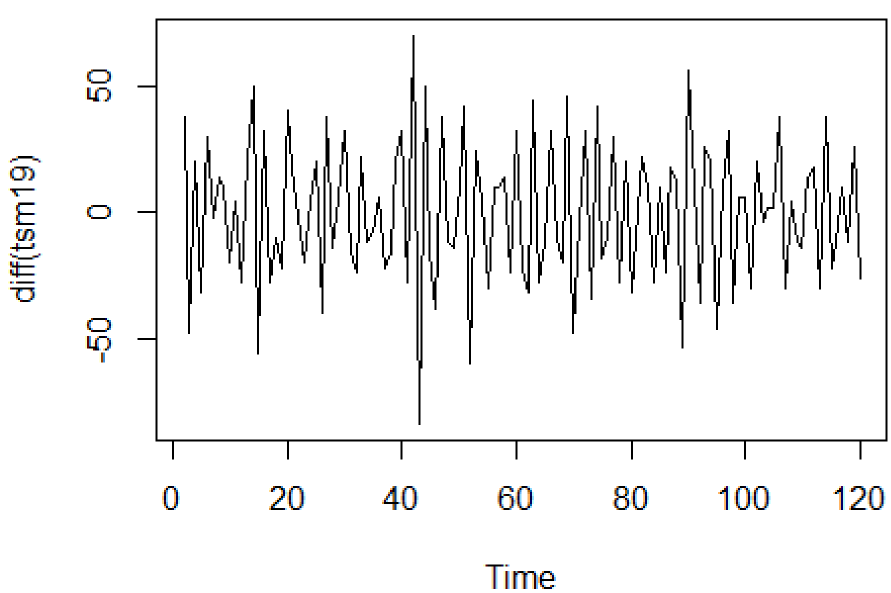
```
milano19 <- milano[year(milano$DATA) == "2019", ]

tsm19 <- ts(milano19$DECESSI)

ts.plot(tsm19)
```



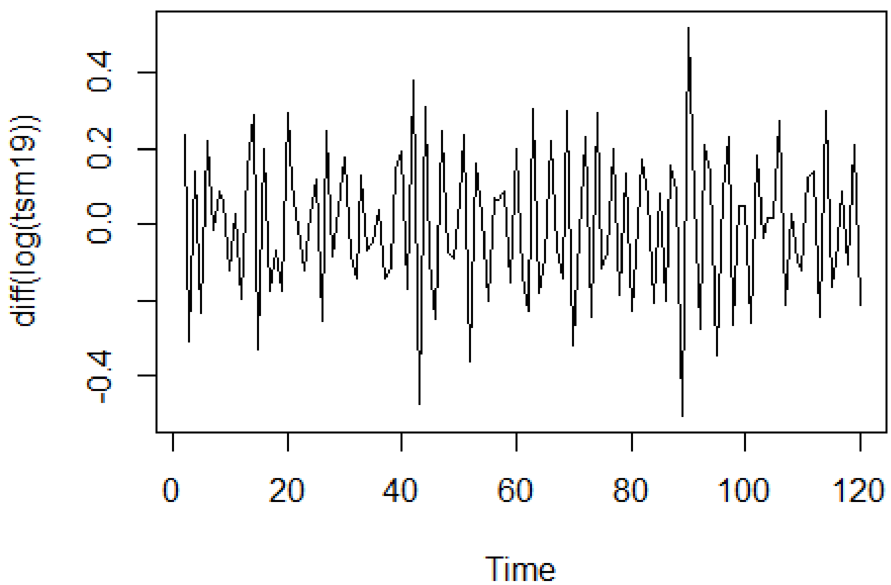

```
ts.plot(log(tsm19))
```




```
ts.plot(diff(tsm19))
```

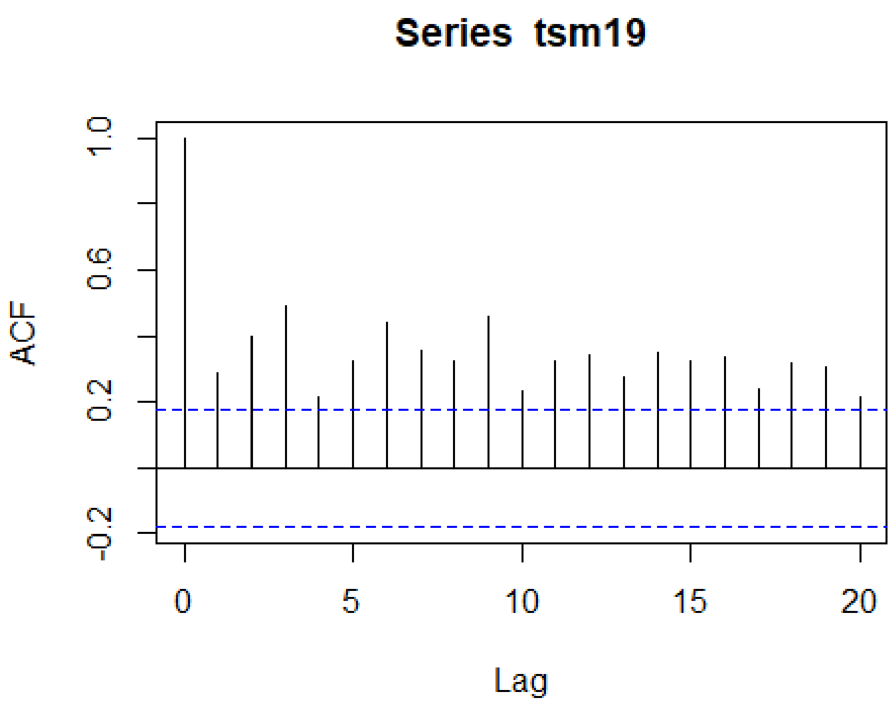



```
ts.plot(diff(log(tsm19)))
```

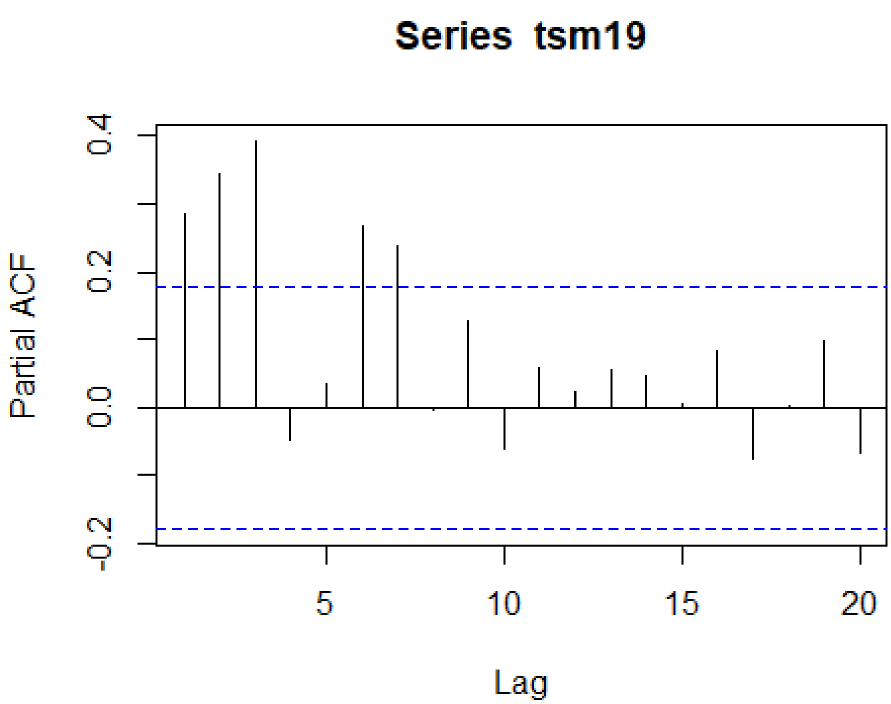


La serie si nota non essere stazionaria in media, la trasformazione logaritmica (che serve a sistemare la non stazionarietà in varianza, ma si fa per prima dato che può sistemare anche quella in media) non sembra risolvere niente. Applicando la differenza prima la questione si sistema un po', si nota un picco a inizio febbraio e inizio aprile. La combo trasformazione + differenza non risolve granchè.

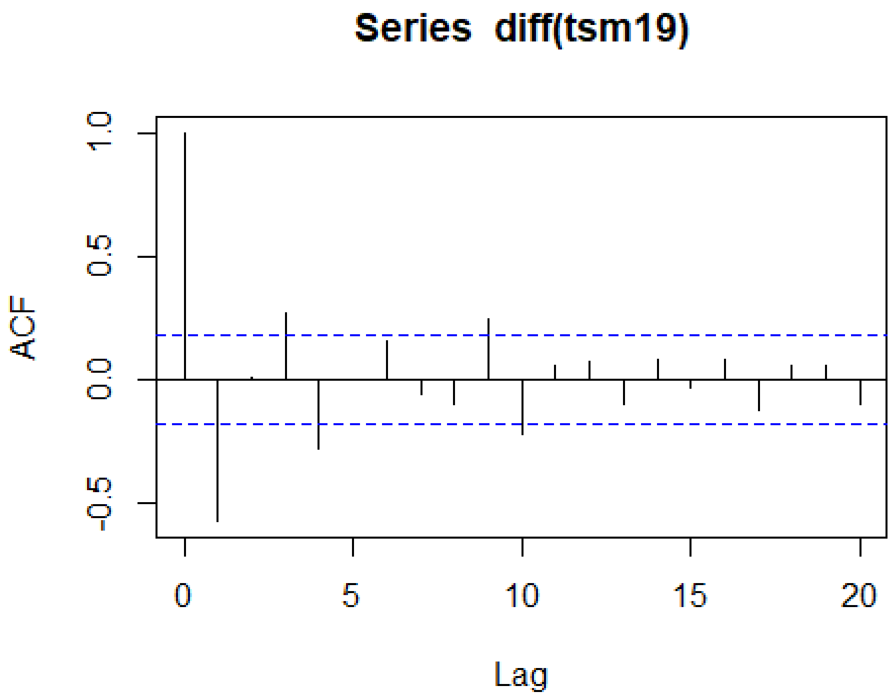
```
acf(tsm19)
```



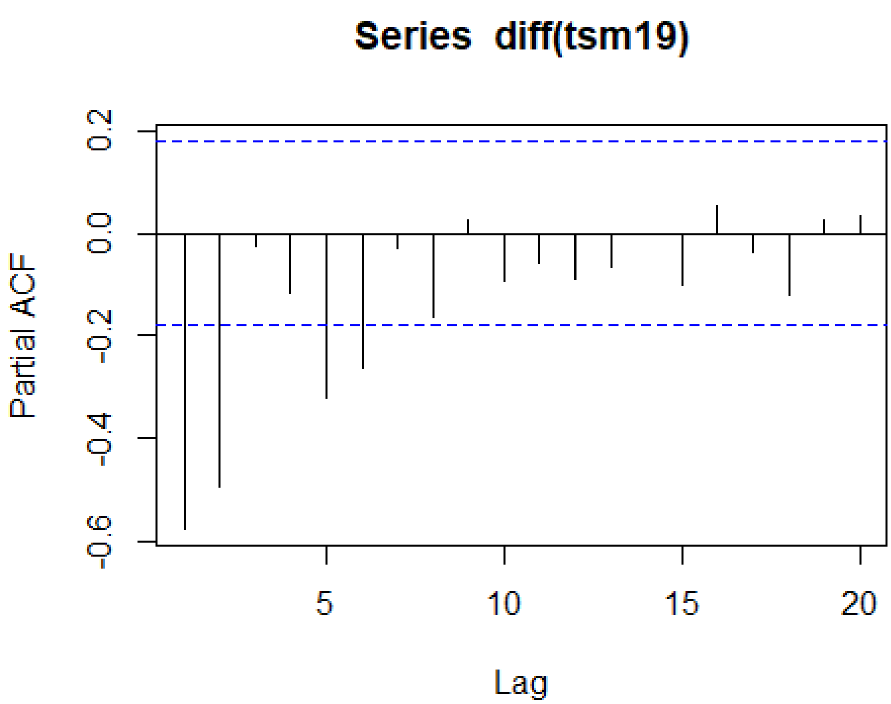

```r
pacf(tsm19)
```




```r
acf(diff(tsm19))
```




```
pacf(diff(tsm19))
```



Andamento strano della pacf che cresce prima di annullarsi. Acf ha andamento stagionale. Con la differenza si vede acf che si annulla al quarto lag con andamento alternato e pacf negativa che si annulla al secondo lag. L'alternarsi delle barrette consiglia il modello AR

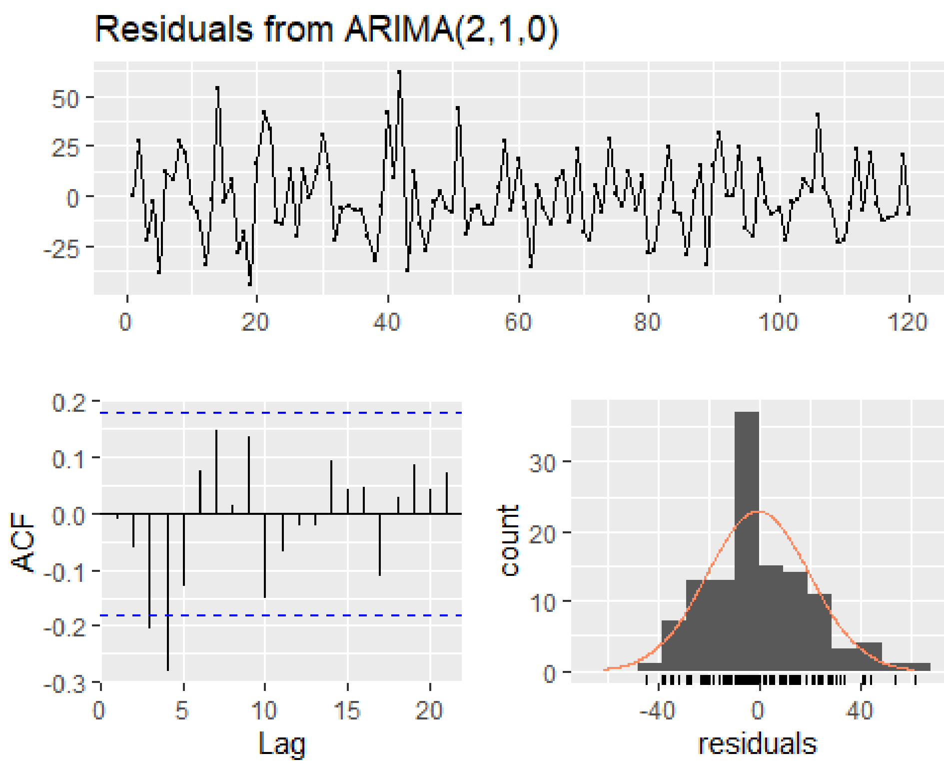
```r
mod7 <- Arima(tsm19, order = c(2, 1, 0))

summary(mod7)
```

```
## Series: tsm19
## ARIMA(2,1,0)
##
## Coefficients:
##           ar1      ar2
##       -0.8630  -0.4937
## s.e.   0.0795   0.0801
##
## sigma^2 estimated as 415.5:  log likelihood=-527.08
## AIC=1060.16   AICc=1060.37   BIC=1068.5
##
## Training set error measures:
##                      ME     RMSE      MAE       MPE     MAPE     MASE
## Training set -0.6334613 20.12733 15.81288 -1.979501 11.02044 0.656571
##                     ACF1
## Training set -0.01146918
```

```r
coeftest(mod7)
```

```
##
## z test of coefficients:
##
##      Estimate Std. Error  z value  Pr(>|z|)
## ar1 -0.862972   0.079479 -10.8578 < 2.2e-16 ***
## ar2 -0.493689   0.080060  -6.1665 6.981e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
checkresiduals(mod7)
```




```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(2,1,0)
## Q* = 27.024, df = 8, p-value = 0.0007003
##
## Model df: 2.   Total lags used: 10
```

```r
#test incorrelazione
LjungBoxTest(residuals(mod7), k = 1, lag.max = 20)
```

```
##    m    Qm       pvalue
##    1  0.02 0.898772313
##    2  0.50 0.479718704
##    3  5.88 0.052949495
##    4 15.75 0.001276035
##    5 17.96 0.001257221
##    6 18.70 0.002186295
##    7 21.52 0.001477248
##    8 21.55 0.003039245
##    9 23.97 0.002317039
##   10 27.02 0.001386057
##   11 27.67 0.002040993
##   12 27.74 0.003548476
##   13 27.80 0.005926186
##   14 28.96 0.006625874
##   15 29.21 0.009796625
##   16 29.51 0.013836095
```

```
##  17 31.26 0.012466082
##  18 31.37 0.018019768
##  19 32.48 0.019284919
##  20 32.76 0.025589885
```

```
#test normalità
jarque.bera.test(residuals(mod7))
```

```
## 
##  Jarque Bera Test
## 
## data:  residuals(mod7)
## X-squared = 4.7602, df = 2, p-value = 0.09254
```

Correlati e normali un po' così così.

#Modello solo per il 2019 con trasformata logaritmica

```
mod8 <- Arima(log(tsm19), order = c(2, 1, 0))

summary(mod8)
```

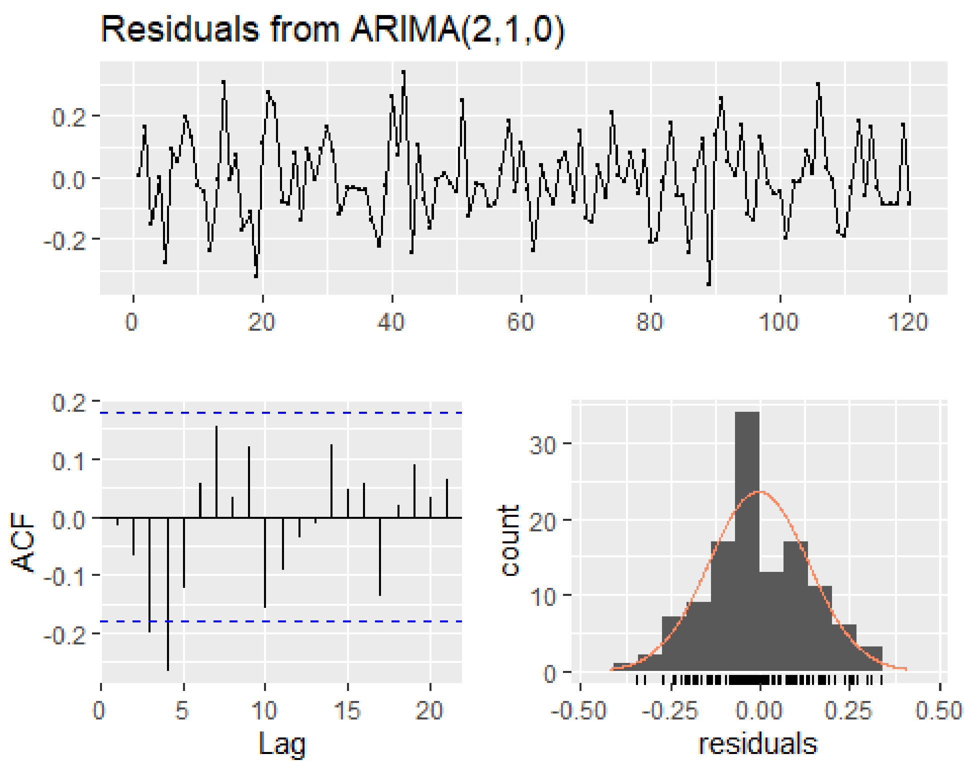
```
## Series: log(tsm19) 
## ARIMA(2,1,0) 
## 
## Coefficients:
##           ar1      ar2
##       -0.8542  -0.5215
## s.e.   0.0780   0.0785
## 
## sigma^2 estimated as 0.0194:  log likelihood=66.23
## AIC=-126.46   AICc=-126.25   BIC=-118.12
## 
## Training set error measures:
##                        ME      RMSE       MAE        MPE     MAPE      MASE
## Training set -0.004711511 0.1375277 0.1083672 -0.1573959 2.186309 0.6513576
##                     ACF1
## Training set -0.0141564
```

```
coeftest(mod8)
```

```
## 
## z test of coefficients:
## 
##      Estimate Std. Error  z value  Pr(>|z|)    
## ar1 -0.854163   0.078010 -10.9494 < 2.2e-16 ***
## ar2 -0.521465   0.078495  -6.6433 3.068e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Come prima AIC e MAPE crollano

```
checkresiduals(mod8)
```




```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(2,1,0)
## Q* = 25.341, df = 8, p-value = 0.001361
##
## Model df: 2.   Total lags used: 10
```

```
#test incorrelazione
LjungBoxTest(residuals(mod8), k = 1, lag.max = 20)
```

```
##    m    Qm      pvalue
##    1  0.02 0.875230524
##    2  0.60 0.437682449
##    3  5.68 0.058396617
##    4 14.52 0.002278462
##    5 16.47 0.002453484
##    6 16.88 0.004725876
##    7 20.00 0.002763730
##    8 20.14 0.005280916
##    9 22.02 0.004878440
##   10 25.34 0.002616454
##   11 26.48 0.003150406
##   12 26.66 0.005173267
##   13 26.67 0.008606155
##   14 28.78 0.007043603
##   15 29.10 0.010116681
##   16 29.56 0.013615938
```

```
##  17 32.19 0.009436337
##  18 32.25 0.014019433
##  19 33.41 0.014893043
##  20 33.58 0.020561302

#test normalità
jarque.bera.test(residuals(mod8))

##
##  Jarque Bera Test
##
## data:  residuals(mod8)
## X-squared = 0.48383, df = 2, p-value = 0.7851
```

Meglio normali, Q si abbassa ma comunque non sono incorrelati. Comunque i grafici non sono terribili

#Riassumendo - Modello 1: ARMA(1,1), AIC = 5303, MAPE = 11, Q = 27.9 con p-val = 0.0002, p-value JB < 0.001 - Modello 2: ARMA(1,1) con trasformazione logaritmica, AIC = -674, MAPE = 2.207, Q = 22.48 con p-val = 0.002, p-value JB = 0.65 - Modello 3: SARMA(1,1)(1,0) con trasformazione logaritmica, AIC = -676.62, MAPE = 2.199, Q = 16.45 con p-value = 0.012, p-value JB = 0.65 - Modello 4: ARIMA(0,1,1) con trasformazione logaritmica, AIC = -665.38, MAPE = 2.22, Q = 21.87 con p-value = 0.0093, JB p-value = 0.62 - Modello 5: SARIMA(0,1,1)(1,0,1) con trasformazione logaritmica, AIC = -668.85, MAPE = 2.20, Q = 15.75 con p-value = 0.027, JB p-value = 0.5566 - Modello 6: Prova di modello con intervento sul 2017, l'intervento non è significativo - Modello 7: Modello solo sul 2019 ARIMA(2,1,0), AIC=1060.16, MAPE = 11.02, Q = 27.02 con p-value = 0.0007, JB p-value = 0.092 - Modello 8: Modello solo sul 2019 ARIMA(2,1,0) con trasformazione logaritmica, AIC = -126.46, MAPE = 2.186, Q = 25.341 con p-value = 0.001, JB p-value = 0.7851

Tutto considerato penso che il modello 2 sia il modello migliore, ma la cosa della correlazione è da studiarsela meglio.