STATISTICA 2

TERZA RELAZIONE

Mazzei Lorenzo

ANALISI

PROBLEMA

Si propone di studiare la serie storica che rappresenta l'andamento della produzione di camion leggeri negli Stati Uniti nel corso degli anni, di modo che si possa capire se il settore ha bisogno di maggiori investimenti a suo supporto o se sta andando già bene. Può inoltre essere interessante capire se esiste un'eventuale stagionalità per vedere i periodi in cui il settore è più fiorente.

DESCRIZIONE DELLA SERIE STORICA

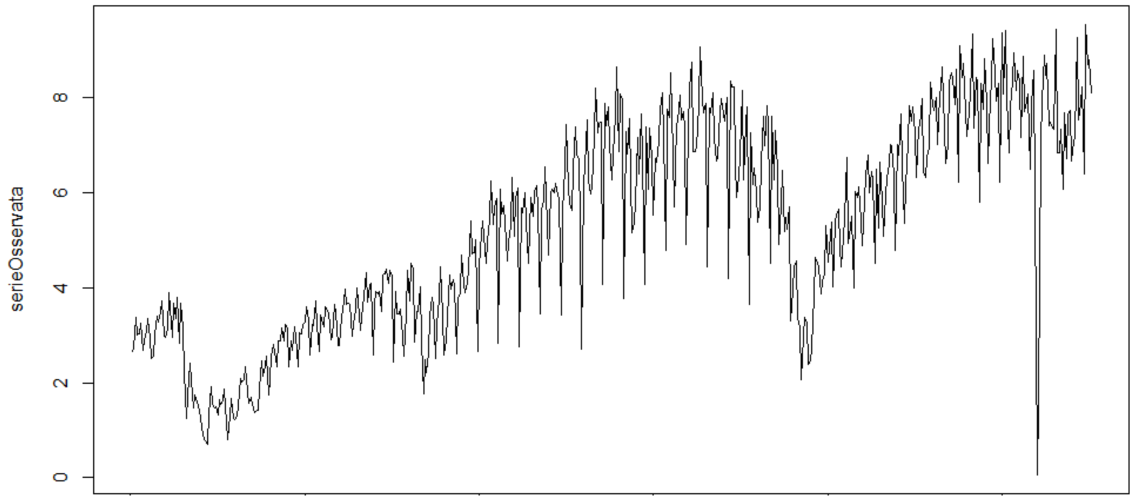
Link tabella: https://fred.stlouisfed.org/series/MVALTTRCKN

La serie storica va dal 1977-01-01 al 2022-11-01. La frequenza dei dati in questione è mensile mentre la loro unità di misura e quindi ciò che compone la serie è la seguente:

- **Milioni di unità**: numero di milioni di camion prodotti mensilmente.

STUDIO INIZIALE

Iniziamo l'analisi con un plot della serie per avere un'idea iniziale dell'andamento dei vari dati:

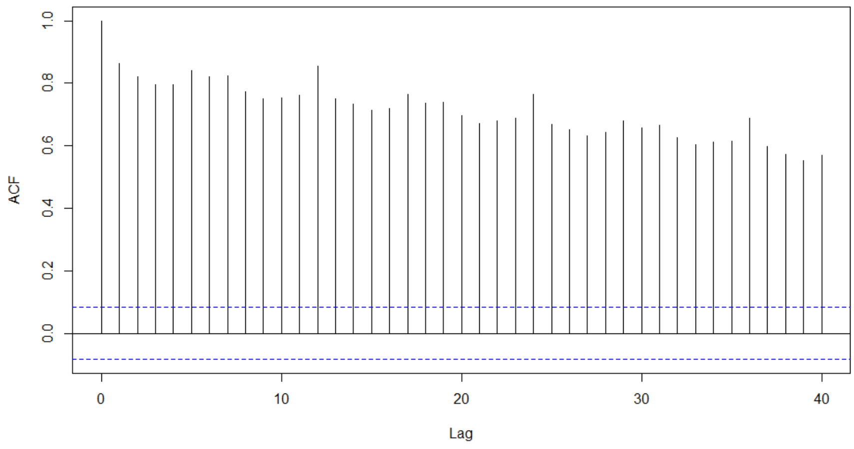


Si può notare che la serie sembra assumere una certa stagionalità con dei picchi di valori verso il basso che si ripetono nel tempo, inoltre a parte per gli anni iniziali sembra esserci anche una componente di trend positivo che però subisce un brusco arresto in corrispondenza del 2008 e del 2020, proviamo a dare una spiegazione a questo fenomeno:

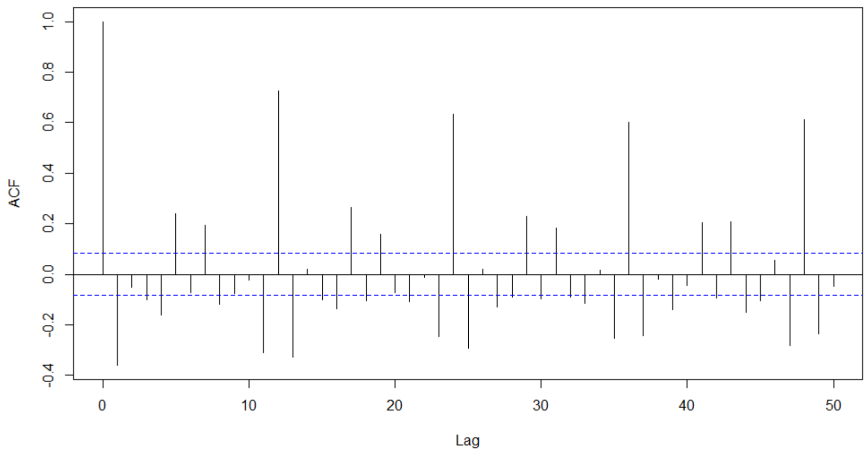
- **2020**: è l'anno in cui è esplosa l'epidemia del Covid, di conseguenza le fabbriche sono state chiuse e il numero di camion prodotti è calato a picco, quasi nullo.
- **2008**: dovuta alla "**automotive industry crisis**" che ha colpito Europa e Asia ma fu sentita soprattutto negli Stati Uniti, che riguarda appunto il nostro caso. La crisi ha portato tra le altre cose anche a un problema di reperimento di materiali per l'assemblaggio (molti non venivano esportati verso gli Stati Uniti per problemi di costo) che spiega il picco in negativo.

AUTOCORRELAZIONI

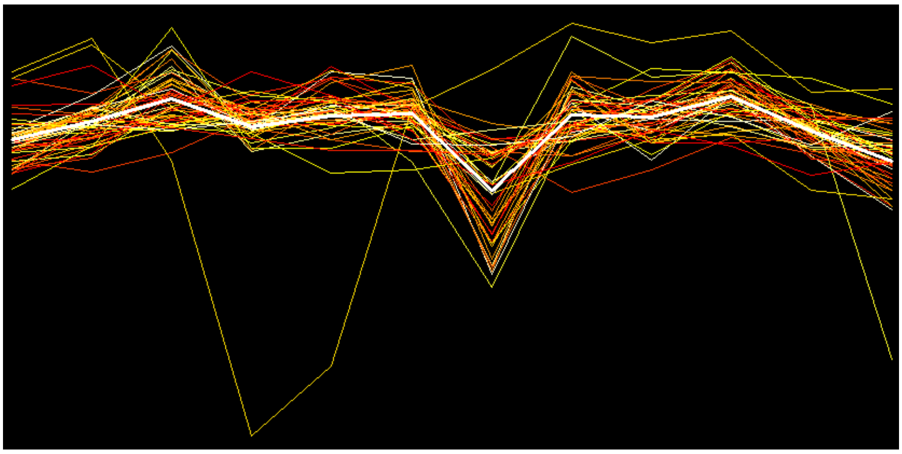
Possiamo usare le autocorrelazioni per cercare di evidenziare un'eventuale presenza di trend e stagionalità come abbiamo ipotizzato nell'analisi iniziale. Di seguito riportato il grafico:



Sembra esserci trend, ma da qui non riusciamo a cogliere in maniera del tutto efficiente la stagionalità anche se l'andamento ondeggiante del grafico suggerisce la presenza di essa. Per discriminarla meglio possiamo differenziare la serie: prendiamo delle differenze per vedere se la serie mostra effettivamente una stagionalità:



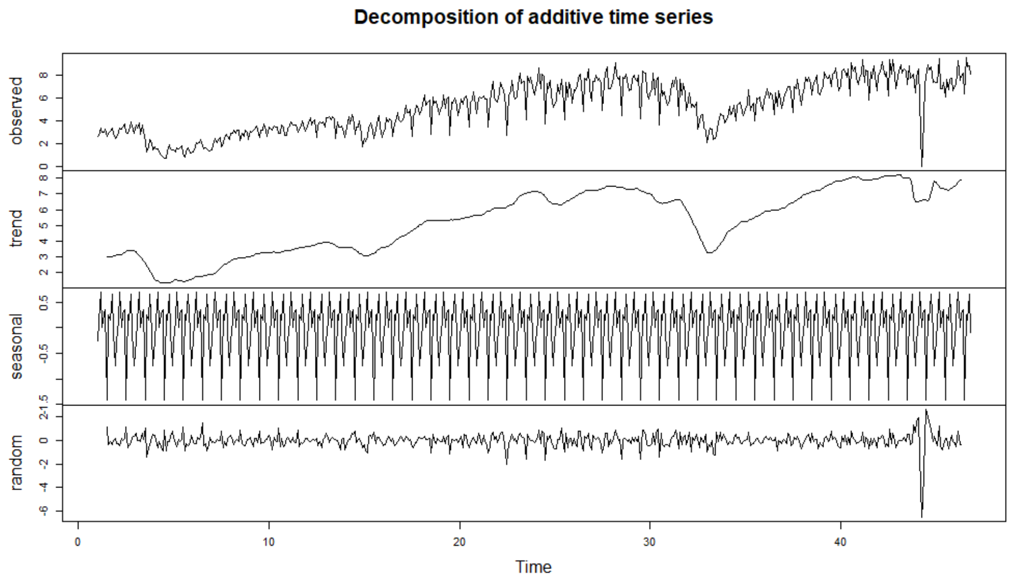
Ora effettivamente emerge la stagionalità, di cui possiamo anche dedurre il periodo tramite il grafico: 12. Possiamo osservare i vari periodi e vedere quanto sono differenti, se al variare dei periodi vediamo figure diverse allora possiamo usare le medie locali sulla stagionalità come decomposizione perché la serie non sarà stazionaria, se invece le figure rimangono uguali allora usiamo la media globale perché la serie sarà stazionaria.



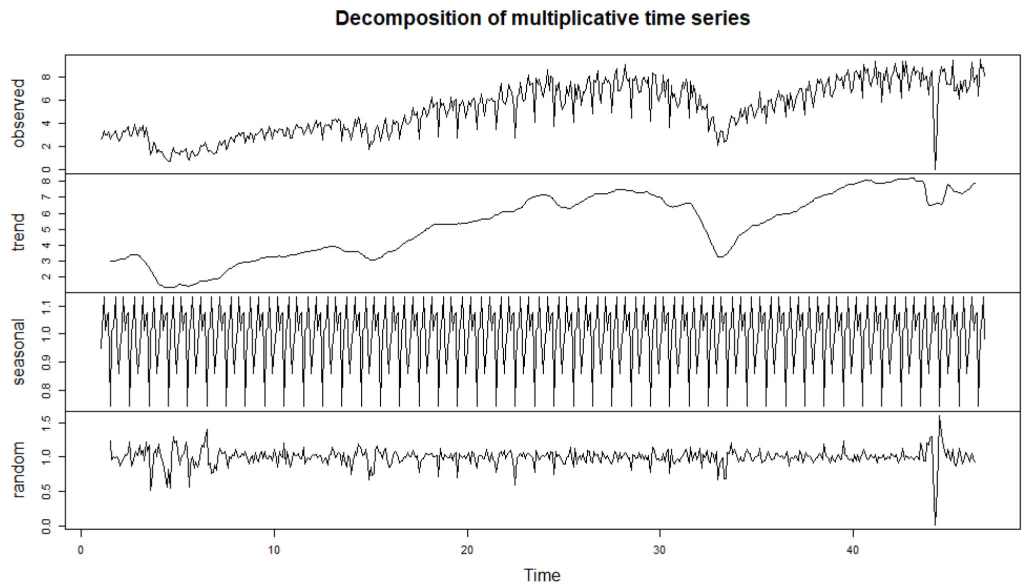
Colori più scuri = anni meno recenti
Colori più chiari = anni più presenti
Bianco = valor medio

La serie sembra essere stazionaria, quindi possiamo studiarla come tale e passare all'analisi delle decomposizioni additive e moltiplicative.

DECOMPOSIZIONI



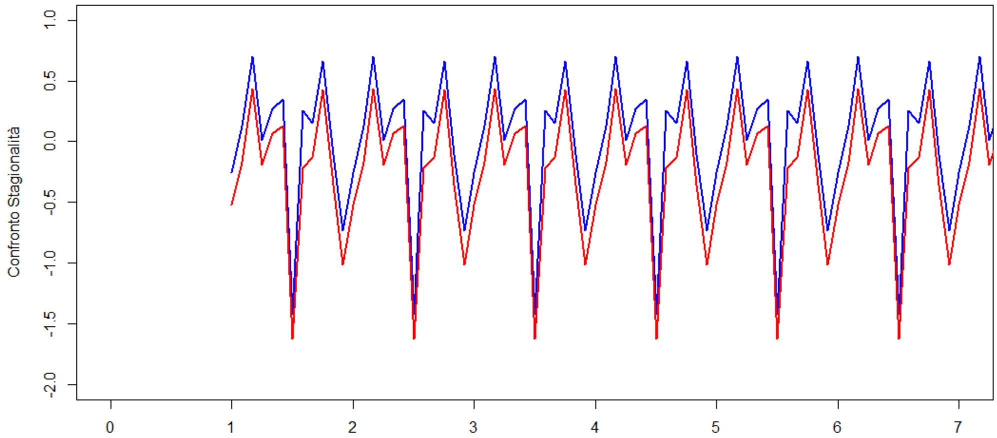
Dalla decomposizione additiva emerge che il trend varia su una scala da 2 a 8 e la stagionalità da -1.5 a 0.5, dunque ha un valore in scala rilevante rispetto al trend. Risulta non banale anche la componente di rumore che, escluso il caso eccezionale del picco negativo, varia da -2 a 2. Sarà necessaria quindi in seguito un'analisi dei residui.



La stagionalità e il rumore non sembrano cambiare molto: gli intervalli di scala per i valori sono più piccoli ma la struttura è simile, unica differenza visiva è la parte iniziale della componente di rumore il cui andamento nel modello moltiplicativo è più accentuato in negativo e in positivo rispetto valore di oscillazione 1.0 a differenza del resto della componente, mentre nel modello additivo mantiene un comportamento abbastanza uniforme.

CONFRONTO STAGIONALITA

Per confrontare le stagionalità dei due modelli dobbiamo fare un passaggio aggiuntivo: serve un valore rappresentativo del trend che permetta di accostare i due andamenti. Nel nostro caso risulta che il valor medio del trend a meno dei valori nulli (NA) dà un risultato abbastanza buono, riportato di seguito:



Blu = stagionalità additiva
Rosso = stagionalità moltiplicativa

In realtà le due stagionalità sono quasi sovrapposte, nel grafico è stata operata una leggera traslazione per far vedere i due andamenti.

Il grafico conferma che la stagionalità è molto simile in entrambi i casi come già ipotizzato in precedenza.

RESIDUI

Per vedere la componente di rumore quanto è rilevante nel problema dobbiamo calcolare la **proporzione di varianza non spiegata** (varianza dei residui rispetto alla varianza dei dati).

- Il computo di questa operazione risulta essere: **0.2692265.**

Essendo il risultato percentuale abbastanza basso, possiamo concludere che la decomposizione che abbiamo utilizzato cattura molto del problema.

Calcoliamo adesso di nuovo la funzione di autocorrelazione: la usiamo come indicatore della presenza di struttura, più le correlazioni sono grandi più vuol dire che c'è struttura nel problema. Idealmente dei residui che sono effettivamente rumore dovrebbero avere valori molto piccoli. Il computo di questa operazione risulta essere:

- Modello **Additivo**: 0.2306728
- Modello **Moltiplicativo**: 0.2281653

OSSERVAZIONE SUL RISULTATO

Sono simili ma il moltiplicativo è più piccolo quindi possiamo pensare che sia più vicino al rumore rispetto alla funzione per il modello additivo. Possiamo concludere allora che c’è una lieve preferenza per il modello moltiplicativo ma sicuramente non netta.

HOLT-WINTERS

Passiamo ora a provare a fare le previsioni della serie. Data la natura della nostra serie possiamo iniziare con uno **smorzamento esponenziale con trend e stagionalità**.

Iniziamo con una “regressione” molto semplice per avere un’idea sulle condizioni iniziali per i valori di “level” e “trend”. Il computo di questa operazione è il seguente: **l.start=2.849,b.start=0.018.** In fase di decomposizione della serie abbiamo concluso che il carattere della serie è stagionale e moltiplicativo, dunque dobbiamo ottimizzare il metodo per α, β, γ. Dopo uno studio con i parametri, una buona soluzione risulta essere:

- **Additivo**: α=0.7, β=0.1, γ=0.3
- **Moltiplicativo**: α=0.3, β=0.1, γ=0.1

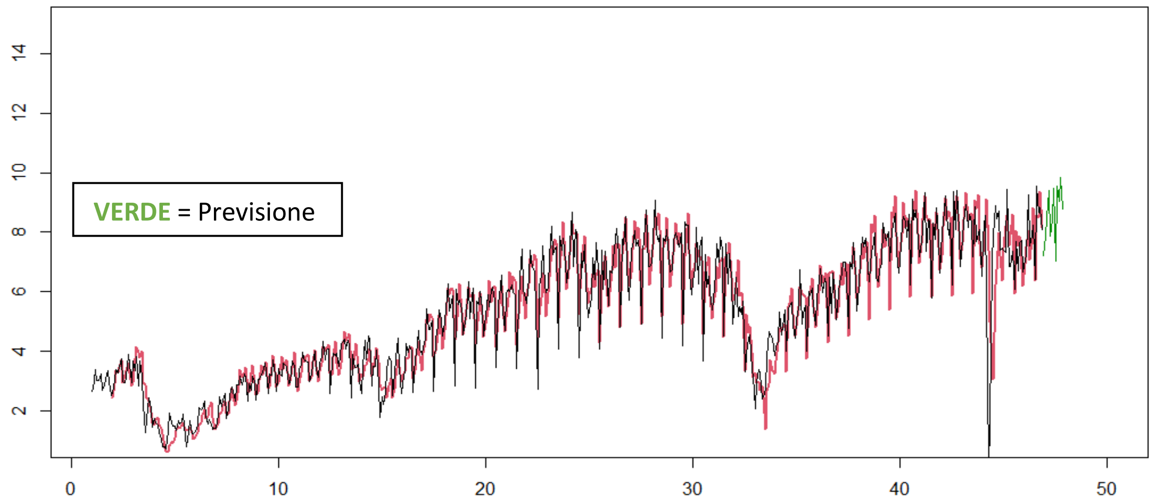
Proviamo a confrontare l’errore che otteniamo dai 3 modelli: quello dove i parametri sono ottimizzati automaticamente, l’additivo e il moltiplicativo:

- **Automatico**: 0.6575488
- **Additivo**: 0.4447068
- **Moltiplicativo**: 0.3979809

Notiamo che in effetti i valori di alpha, beta e gamma scelti da noi garantiscono un errore minore. Confrontiamo i due modelli per validazione e guardiamo la misura dell’errore in previsione:

- **Additivo**: 10.43538
- **Moltiplicativo**: 8.779849

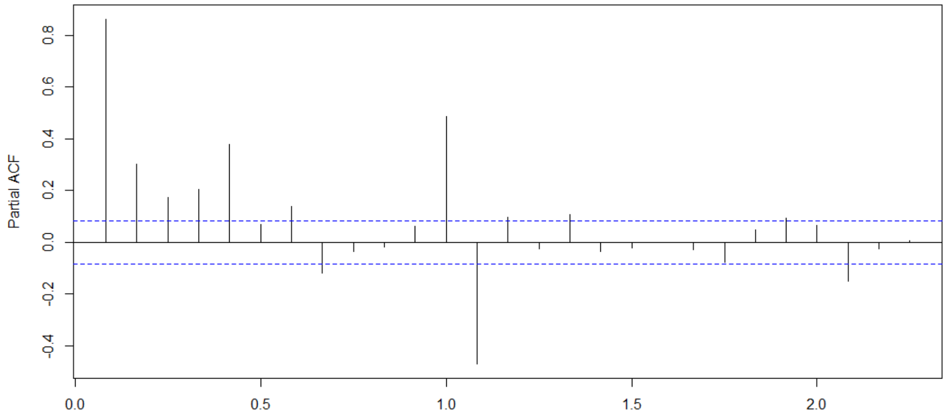
Dalle analisi condotte possiamo concludere che il modello moltiplicativo sia quello migliore, quindi lo usiamo per fare una previsione, insieme ad una stima non parametrica dell’incertezza:



METODI AUTOREGRESSIVI

Passiamo ora all'analisi dei metodi regressivi sempre con lo scopo di ottenere una previsione finale per poi confrontare questi modelli col risultato di Holt-Winters ottenuto in precedenza e capire quale sia il migliore.

Il grafico dell'autocorrelazione parziale porta a ipotizzare una dipendenza che arriva fino a 8 lag o poco più:



Dato il periodo 12, realizziamo l'autoregressione con un modello implementato direttamente usando 13 lag (quindi 13 colonne d'ingresso e la quattordicesima di uscita).

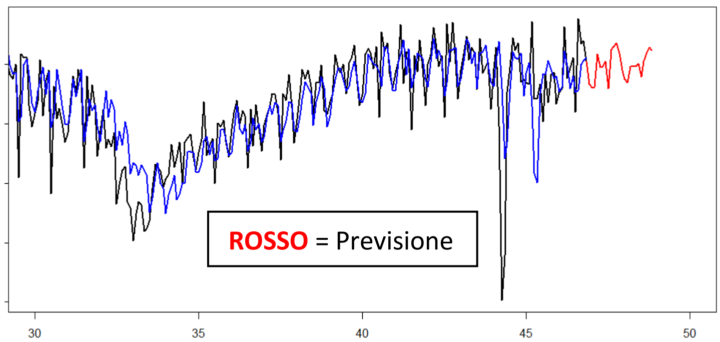
Come si può vedere dalle due immagini sottostanti, molti coefficienti hanno un p-value non banale, quindi possiamo pensare di ridurre il modello. Eliminando i coefficienti con p-value più elevato aumenta la varianza spiegata di 0.01, mentre sottraendo solo i 2 coefficienti con p-value più basso otteniamo una varianza spiegata minore ma non in maniera significativa (0.05).

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.13014    0.08367   1.555  0.12045
X1          -0.47649    0.03894 -12.236  < 2e-16 ***
X2           0.69443    0.03583  19.383  < 2e-16 ***
X3          -0.09606    0.03560  -2.698  0.00719 **
X4          -0.05664    0.03550  -1.595  0.11123
X5          -0.01546    0.03555  -0.435  0.66368
X6           0.00330    0.03631   0.091  0.92763
X7           0.04031    0.03626   1.112  0.26676
X8           0.02423    0.03628   0.668  0.50444
X9           0.18599    0.03532   5.265 2.05e-07 ***
X10          0.03495    0.03519   0.993  0.32115
X11          0.01689    0.03517   0.480  0.63131
X12          0.03730    0.03535   1.055  0.29186
X13          0.58754    0.03853  15.248  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.6916 on 524 degrees of freedom
Multiple R-squared:  0.9033,    Adjusted R-squared:  0.9009
F-statistic: 376.4 on 13 and 524 DF,  p-value: < 2.2e-16
```

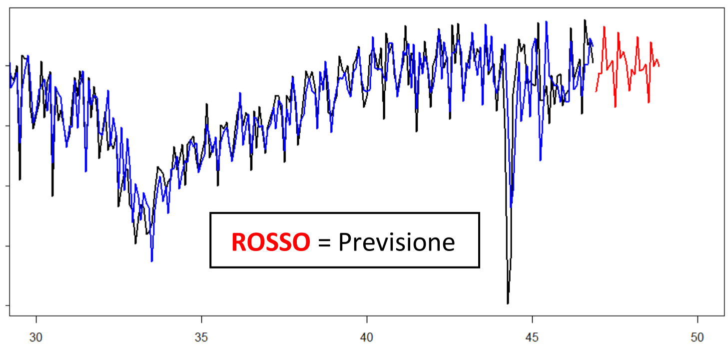
```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.16738    0.09900   1.691   0.0915 .
X2           0.54175    0.02696  20.092   <2e-16 ***
X13          0.43889    0.02678  16.387   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.8431 on 535 degrees of freedom
Multiple R-squared:  0.8532,    Adjusted R-squared:  0.8527
F-statistic:  1555 on 2 and 535 DF,  p-value: < 2.2e-16
```

Proseguiamo quindi facendo le previsioni a mano e confrontandole:

MODELLO RIDOTTO

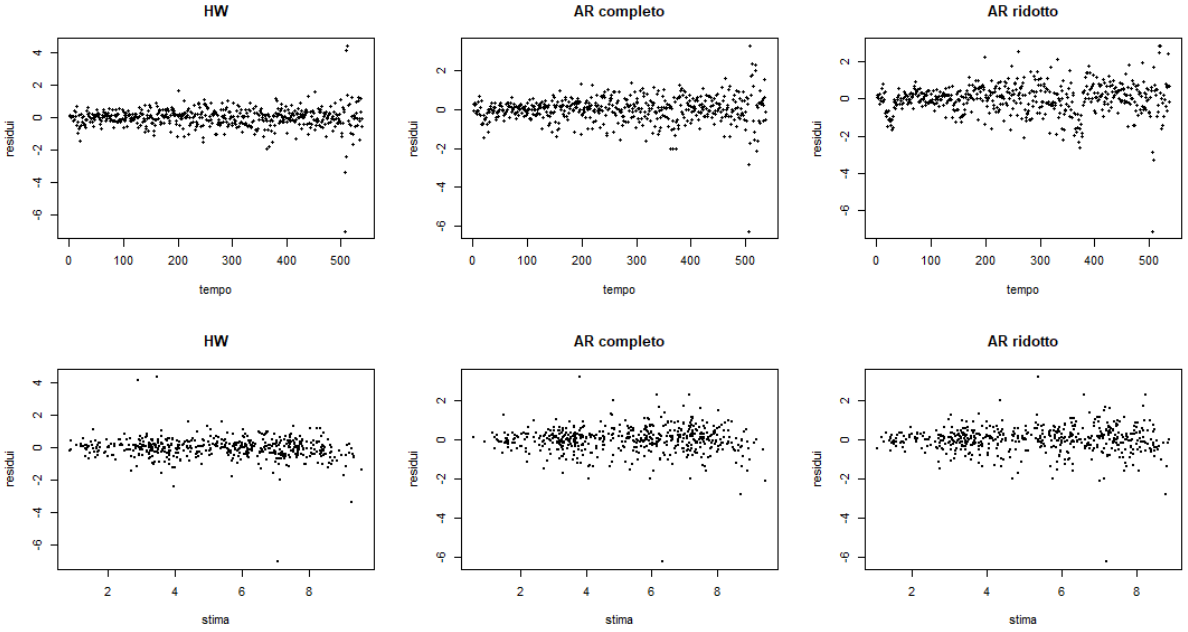


MODELLO COMPLETO



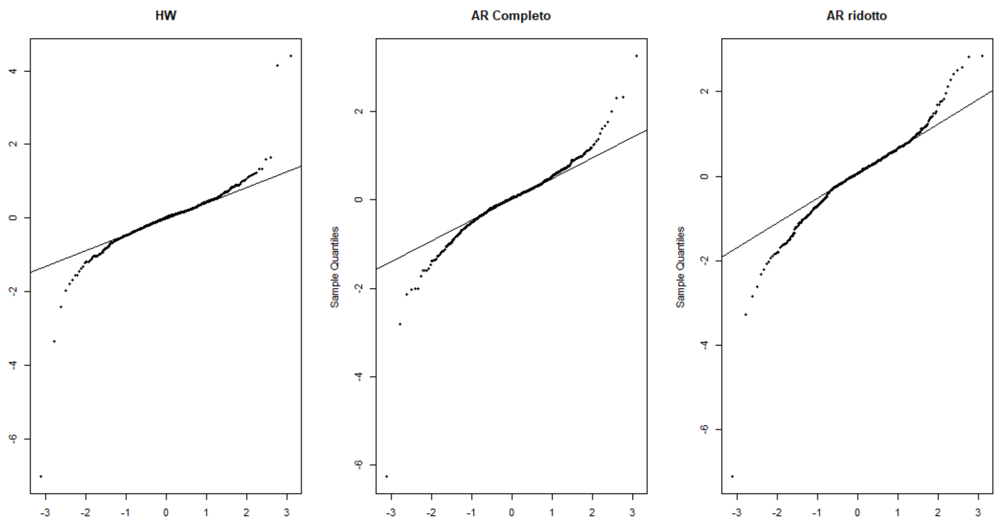
OSSERVAZIONI MEDIANTE GRAFICI

Grafici dei residui:



In HW sembra esserci più coesione dei residui, nell'AR completo questa risulta minore, il peggiore sembrerebbe essere l'AR ridotto. Dai grafici dell'autocorrelazione parziale (qui non riportati per pulizia grafica) sembra mantenersi la stagionalità).

Grafici dei Quantile:



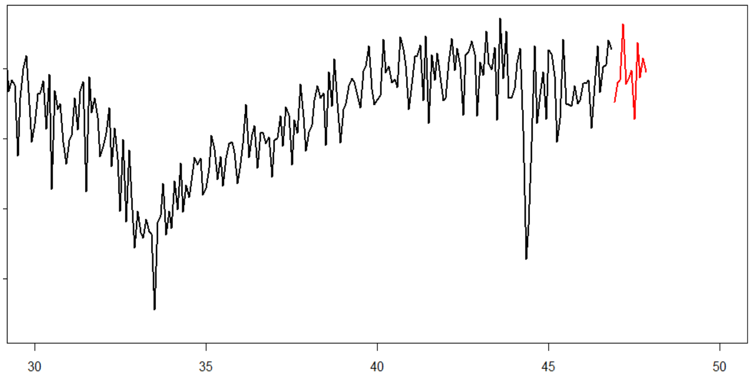
Sicuramente osservando i quantili il modello peggiore sembra essere l'AR ridotto. Tra i due rimanenti quello che sembra distaccarsi di meno (e quindi il migliore) è HW.

Confronto errori

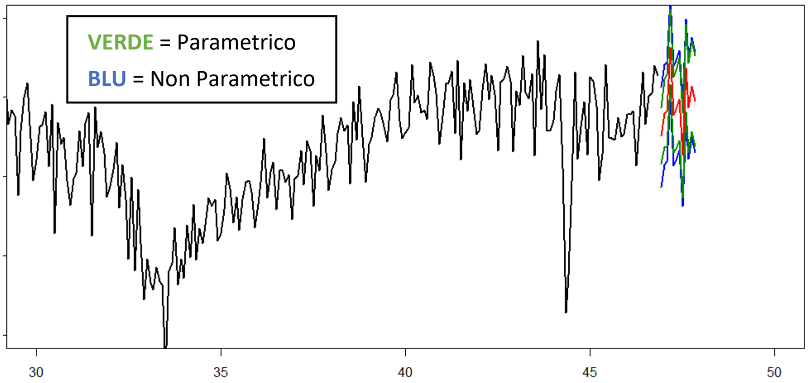
Holt-Winters = 0.7045752, **Completo** = 1.054663, **Ridotto** = 1.298767. Dunque anche gli errori sembrano suggerire che il metodo migliore sia Holt-Winters, in linea con quanto avevamo ipotizzato con i grafici dei residui e dei quantili. Ridurre il modello quindi non ha portato a risultati soddisfacenti.

YULE-WALKER

PREDIZIONE

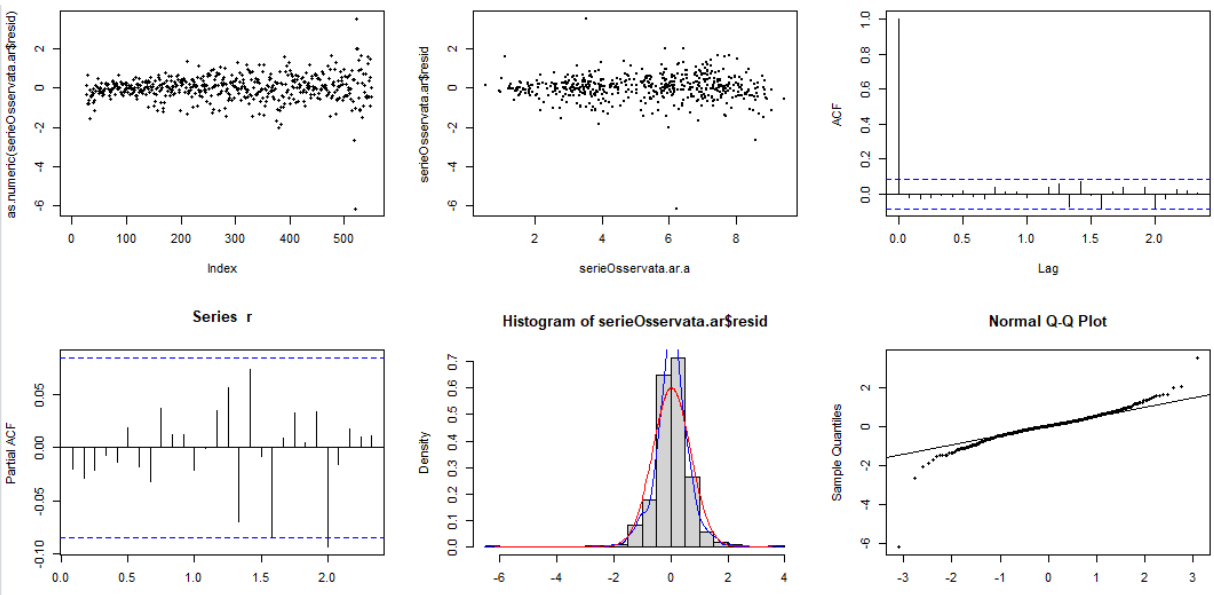


STIMA INCERTEZZA



OSSERVAZIONI MEDIANTE GRAFICI

Proviamo a vedere se possiamo trarre delle osservazioni mediante alcuni grafici, in particolare concentrandoci sui residui e i quantili come fatto per il modello precedente:

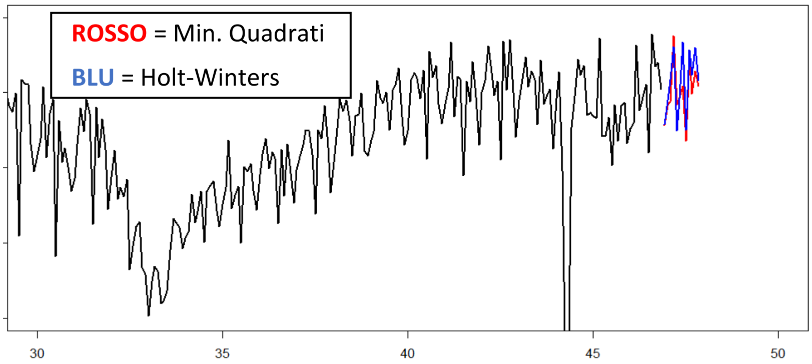


I residui sembrano avere abbastanza coesione almeno nella prima parte, poi tendono a disperdersi. Il quantile sembra nella media: i valori centrali non si discostano molto mentre gli estremi si staccano in modo abbastanza evidente. In generale comunque il risultato appare abbastanza buono.

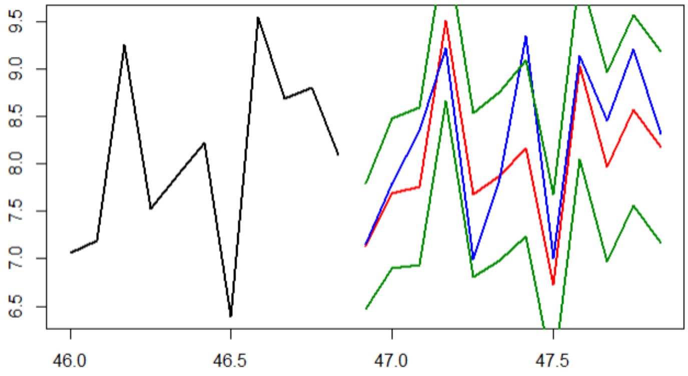
Confronto errori: **Holt-Winters** = 2.655595, **Yule-Walker** = 0.5301485. Yule-Walker dunque risulta preferibile rispetto a Holt-Winters.

MINIMI QUADRATI

Confronto con Holt-Winters

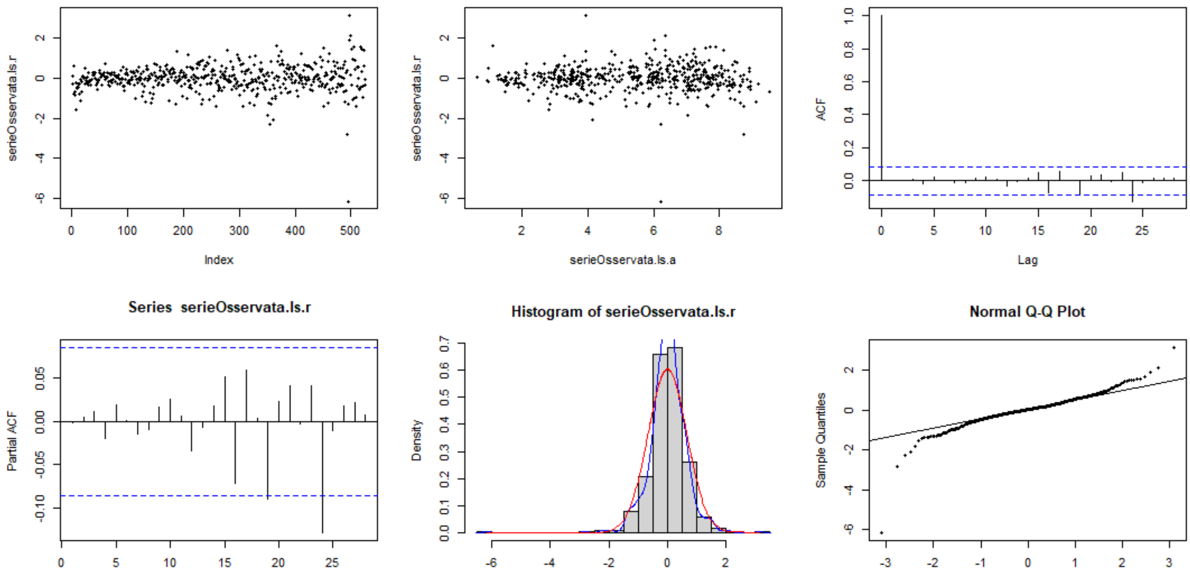


ZOOM



OSSERVAZIONI MEDIANTE GRAFICI

Come fatto per Yule-Walker utilizziamo i grafici per trarre delle osservazioni:



Per i residui si possono fare bene o male le stesse considerazioni di Yule-Walker: più coesione nella prima parte, poi tendono a disperdersi. Anche Il quantile si comporta come osservato in precedenza: i valori centrali hanno un buon andamento mentre gli estremi si discostano in maniera non banale.

Confronto errori: **Holt-Winters** = 19.49897, **Minimi Quadrati** = 1.157639. Il metodo dei Minimi quadrati risulta essere significativamente migliore rispetto a Holt-Winters.

CONCLUSIONE

Studiando la serie abbiamo riconosciuto la sua stagionalità e stabilito che la decomposizione moltiplicativa sia la migliore. Per quanto riguarda i metodi di previsione: in Holt-Winters il modello moltiplicativo risulta essere quello più indicato per fare una stima. Studiando i metodi autoregressivi dalle analisi dei grafici e dai confronti degli errori abbiamo visto che Holt-Winters è migliore rispetto al modello fatto a mano ma peggiore rispetto a Yule-Walker e Minimi Quadrati che dunque possiamo usare per fare delle stime.

APPENDICE

#eliminazione fattore ‘cost’; plot della serie e autocorrelazione

```
class(serieOriginale)

serieOsservata = ts(serieOriginale[,'cost'])

plot(serieOsservata)

ts.plot(serieOsservata)

acf(serieOsservata,40)

acf(diff(serieOsservata),50)

par(bg="black")

m_ap=matrix(serieOsservata,12,45)

#ts.plot(m_ap,col=heat.colors(12))

ts.plot(scale(m_ap,scale=F),col=heat.colors(12))

lines(rowMeans(scale(m_ap,scale=F)),lwd=3,col="white")

par(bg="white")
```

#Decomposizione

```
serieOsservata=ts(serieOsservata,frequency=12)

serieOsservata.decomposeAdditiva = decompose(serieOsservata)

serieOsservata.decomposeMoltiplicativa = decompose(serieOsservata,type =
"multiplicative")

layout(matrix(1:2,1,2))

plot(serieOsservata.decomposeAdditiva)

plot(serieOsservata.decomposeMoltiplicativa)

layout(1)
```

#Confronto Stagionalità

```
mediaTrend = mean(serieOsservata.decomposeMoltiplicativa$trend,na.rm = TRUE)

plot(serieOsservata.decomposeAdditiva$seasonal,col = "blue",xlim=c(0,7),ylab =
"Confronto Stagionalità", ylim = c(-2,1) ,lwd = 2)

lines(mediaTrend*(serieOsservata.decomposeMoltiplicativa$seasonal-
1.05),col="red",lwd = 2)
```

#Residui

```
serieOsservataMoltiplicativa.residui =
na.exclude(serieOsservata.decomposeMoltiplicativa$random)

serieOsservataMoltiplicativa.residui = log(serieOsservataMoltiplicativa.residui)
```

```
plot(serieOsservataMoltiplicativa.residui)

acf(serieOsservataMoltiplicativa.residui)

serieOsservata.decomposeMoltiplicativa$random

var(serieOsservataMoltiplicativa.residui)/var(window(log(serieOsservata),c(1,7),
c(46,5)))

sd(acf(serieOsservataAdditiva.residui,plot = F)$acf)

sd(acf(na.exclude(serieOsservata.decomposeMoltiplicativa$random),plot = F)$acf)
```

#Osservazione su stagionalità non periodica

```
plot(stl(serieOsservata,7))

plot(stl(serieOsservata,"periodic"))
```

#Holt-Winters

```
x=1:20

coefficients(lm(serieOsservata[1:20]~x))

serieOsservata.hwa =
HoltWinters(serieOsservata,alpha=0.7,beta=0.1,gamma=0.3,seasonal = "additive")

plot(serieOsservata.hwa, lwd = 2)

serieOsservata.hwm =
HoltWinters(serieOsservata,alpha=0.3,beta=0.1,gamma=0.1,seasonal =
"multiplicative")

plot(serieOsservata.hwm, lwd = 2)

nt=18 # numero di test set

ft=1 # unità di tempo nel futuro su cui valutare la previsione

n=length(serieOsservata) # numero totale di anni

idt=start(serieOsservata) # data di inizio della serie

fdt=end(serieOsservata)   # data di fine della serie

pdt=frequency(serieOsservata) # periodo della serie

err_hw0=0

err_hw1=0

err_hw2=0

for(j in (n-nt-ft):(n-ft-1)){

  # costruzione di train e test

  train=window(serieOsservata,idt,ts_data(idt,pdt,j))

  future=ts_data(idt,pdt,j+ft)

  test=window(serieOsservata,future,future)
```

```
  # HW standard
  train.hw0=HoltWinters(train)
  err_hw0=err_hw0+sum((as.numeric(test)-as.numeric(predict(train.hw0,ft)))^2)
  # HW parametri personalizzati
  train.hw1=HoltWinters(train,alpha=0.7, beta=0.1, gamma=0.3, l.start=2.849,
b.start=0.018,seasonal = "additive")
  err_hw1=err_hw1+sum((as.numeric(test)-as.numeric(predict(train.hw1,ft)))^2)
  train.hw2=HoltWinters(train,alpha=0.3, beta=0.1, gamma=0.1, l.start=2.849,
b.start=0.018,seasonal = "multiplicative")
  err_hw2=err_hw2+sum((as.numeric(test)-as.numeric(predict(train.hw2,ft)))^2)
}
err_hw0/nt
err_hw1/nt
err_hw2/nt

nt=20 # numero di test set
ft=1 # unità di tempo nel futuro su cui valutare la previsione
n=length(serieOsservata) # numero totale di anni
idt=start(serieOsservata) # data di inizio della serie
fdt=end(serieOsservata)   # data di fine della serie
pdt=frequency(serieOsservata) # periodo della serie

err_a=0
err_m=0
for(l in (n-nt-ft):(n-1-ft)){
  # costruzione di train e test
  train=window(serieOsservata,idt,ts_data(idt,pdt,l))
  future=ts_data(idt,pdt,l+ft)
  test=window(serieOsservata,future,future)
  train.hwa=HoltWinters(train,alpha=0.7,beta=0.1,gamma=0.3,seasonal="additive")
  train.hwm=HoltWinters(train,alpha=0.3
,beta=0.1,gamma=0.1,seasonal="multiplicative")
  err_a = err_a + (as.numeric(test) - as.numeric(predict(train.hwa,ft)))^2
```

```
  err_m = err_m + (as.numeric(test) - as.numeric(predict(train.hwm,ft)))^2
}
err_a
err_m

serieOsservata.hwm.p=predict(serieOsservata.hwm,12)
serieOsservata.hwm.r=resid(serieOsservata.hwm)
plot(serieOsservata.hwm,lwd=2,xlim = c(1,50),ylim = c(1,15))
lines(serieOsservata.hwm.p,col="green4")
```

#Metodi Regressivi (modello fatto da noi)

```
acf(serieOsservata)
pacf(serieOsservata)
L = length(serieOsservata)
l = 13  # numero di lag in ingresso
mnt = matrix(nrow = L - l, ncol = l + 1)
for (i in 1:(l + 1)) {
    mnt[, i] = serieOsservata[i:(L - l - 1 + i)]
}
mnt <- data.frame(mnt)
nt.lm <- lm(X14 ~ ., data = mnt)  # X14 perché 13 lag in ingresso
summary(nt.lm)
copia = mnt[,-c(6,5,11,8)]
nt.lmr <- lm(X14 ~ X1+X2+X3+X4+X7+X9+X10+X12+X13, data = mnt)
summary(nt.lmr)
nt.lmr <- lm(X14 ~ X2+X13, data = mnt)
summary(nt.lmr)
```

#ridotto

```
anni = 2 #periodi che vogliamo stimare
L = length(serieOsservata)
ptr = rep(0, L + 12 * anni)
ptr[1:L] = serieOsservata
for (i in 1:(12 * anni)) {
```

```r
    ptr[L + i] = coef(nt.lmr) %*% c(1, ptr[L + i - 12], ptr[L + i - 1])
}
nt.lmr.pt = ts(ptr, frequency = 12, start = c(1, 1))
nt.lmr.a = window(serieOsservata,c(2,2)) - resid(nt.lmr)
ts.plot(serieOsservata, nt.lmr.a, window(nt.lmr.pt,c(46, 11)), col =
c("black","blue","red"),xlim = c(30,50),lwd = 2)
#completo
nt.lmc <- lm(X14 ~ ., data = mnt)
anni = 2
ptc = rep(0, L + 12 * anni)
ptc[1:L] = serieOsservata.ts
for (i in 1:(12 * anni)) {
    ptc[L + i] = coef(nt.lmc) %*% c(1, rev(ptc[L + i - 1:l]))
}
nt.lmc.pt = ts(ptc, frequency = 12, start = c(1, 1))
nt.lmc.a = window(serieOsservata.ts, c(2, 2)) - resid(nt.lmc)
ts.plot(serieOsservata.ts, nt.lmc.a, window(nt.lmc.pt, c(46, 12)), col =
c("black",
    "blue", "red"),xlim = c(30,50),lwd = 2)
#analisi residui
# estrazione dei residui
nt.hw.r = resid(nt.hw)
nt.lmc.r = resid(nt.lmc)
nt.lmr.r = resid(nt.lmr)
# varianze non spiegate
var(nt.hw.r)/var(window(serieOsservata, 2))
var(nt.lmc.r)/var(window(serieOsservata, 2))
var(nt.lmr.r)/var(window(serieOsservata, 2))
length(nt.lmc.a)
length(nt.lmc.r)
# indicatori di forma
fm = matrix(c(s2_skewness(nt.hw.r), s2_skewness(nt.lmc.r),
s2_skewness(nt.lmr.r),
```

```
    s2_kurtosis(nt.hw.r), s2_kurtosis(nt.lmc.r), s2_kurtosis(nt.lmr.r)),
    3, 2)
colnames(fm) <- c("skewness", "kurtosi")
rownames(fm) <- c("HoltWinters", "autoreg. completo", "autoreg. ridotto")
fm
# confronto grafico
layout(matrix(1:6, 2, 3, byrow = T))
plot(as.numeric(nt.hw.r), pch = 20, main = "HW", xlab = "tempo", ylab =
"residui")
plot(nt.lmc.r, pch = 20, main = "AR completo", xlab = "tempo", ylab = "residui")
plot(nt.lmr.r, pch = 20, main = "AR ridotto", xlab = "tempo", ylab = "residui")
plot(nt.hw$fitted[, 1], nt.hw.r, pch = 20, main = "HW", xlab = "stima", ylab =
"residui")
plot(nt.lmc.a, nt.lmc.r, pch = 20, main = "AR completo", xlab = "stima",
    ylab = "residui")
plot(nt.lmr.a, nt.lmc.r, pch = 20, main = "AR ridotto", xlab = "stima", ylab =
"residui")
layout(1)
# acf e pacf
layout(matrix(1:6, 2, 3, byrow = T))
acf(nt.hw.r, 28)
acf(nt.lmc.r, 28)
acf(nt.lmr.r, 28)
pacf(nt.hw.r, 28)
pacf(nt.lmc.r, 28)
pacf(nt.lmr.r, 28)
layout(1)
# frequenze
layout(t(1:3))
hist(nt.hw.r, 20, freq = F, main = "HW")
lines(density(nt.hw.r), col = "blue")
lines(sort(nt.hw.r), dnorm(sort(nt.hw.r), mean(nt.hw.r), sd(nt.hw.r)), col =
"red")
hist(nt.lmc.r, 20, freq = F, main = "AR completo")
```

```r
lines(density(nt.lmc.r), col = "blue")
lines(sort(nt.lmc.r), dnorm(sort(nt.lmc.r), mean(nt.lmc.r), sd(nt.lmc.r)),
    col = "red")
hist(nt.lmr.r, 20, freq = F, main = "AR ridotto")
lines(density(nt.lmr.r), col = "blue")
lines(sort(nt.lmr.r), dnorm(sort(nt.lmr.r), mean(nt.lmr.r), sd(nt.lmr.r)),
    col = "red")
layout(1)
# quantili
layout(t(1:3))
qqnorm(nt.hw.r, pch = 20,main = "HW")
qqline(nt.hw.r)
qqnorm(nt.lmc.r, pch = 20,main = "AR Completo")
qqline(nt.lmc.r)
qqnorm(nt.lmr.r, pch = 20,main = "AR ridotto")
qqline(nt.lmr.r)
layout(1)
shapiro.test(nt.hw.r)
shapiro.test(nt.lmc.r)
shapiro.test(nt.lmr.r)
layout(1)
#confronto errori
nt = 20  # numero di test set
ft = 1  # unità di tempo nel futuro su cui valutare la previsione
n = length(serieOsservata)  # numero totale di anni
idt = start(serieOsservata)  # data di inizio della serie
fdt = end(serieOsservata)  # data di fine della serie
pdt = frequency(serieOsservata)  # periodo della serie

err_hw = rep(0, nt)
err_lmc = rep(0, nt)
err_lmr = rep(0, nt)
```

```
for (j in (n - nt - ft):(n - ft - 1)) {
    # training e test set
    train = window(serieOsservata, idt, ts_data(idt, pdt, j))
    future = ts_data(idt, pdt, j + ft)
    test = window(serieOsservata, future, future)
    # HW
    train.hw = HoltWinters(train)
    err_hw[j - (n - nt - ft) + 1] = as.numeric(test) - predict(train.hw,
        ft)[ft]
    # AR
    L = length(train)
    l = 13  # numero di lag in ingresso
    mtrain = matrix(nrow = L - l, ncol = l + 1)
    for (i in 1:(l + 1)) {
        mtrain[, i] = train[i:(L - l - 1 + i)]
    }
    mtrain <- data.frame(mtrain)
    # AR completo
    train.lmc <- lm(X14 ~ ., data = mtrain)
    train.lmc.p = rep(0, L + ft)
    train.lmc.p[1:L] = train
    for (i in 1:ft) {
        train.lmc.p[L + i] = coef(train.lmc) %*% c(1, rev(train.lmc.p[L +
            i - 1:l]))
    }
    err_lmc[j - (n - nt - ft) + 1] = as.numeric(test) - train.lmc.p[L + ft]
    # AR ridotto
    train.lmr <- lm(X14 ~ X3 + X13, data = mtrain)
    train.lmr.p = rep(0, L + ft)
    train.lmr.p[1:L] = train
    for (i in 1:ft) {
        train.lmr.p[L + i] = coef(train.lmr) %*% c(1, train.lmr.p[L + i -
```

```
            11], train.lmr.p[L + i - 1])
    }
    err_lmr[j - (n - nt - ft) + 1] = as.numeric(test) - train.lmr.p[L + ft]
}
sum(err_hw^2)/nt
sum(err_lmc^2)/nt
sum(err_lmr^2)/nt
```

#Yule-Walker

```
serieOsservata.ar = ar(serieOsservata)
ts.plot(serieOsservata, serieOsservata - serieOsservata.ar$resid, col = 
c("black", "red"))
r = na.omit(serieOsservata.ar$resid)
serieOsservata.ar.pt = predict(serieOsservata.ar, n.ahead = 12, se.fit = TRUE, 
level = 0.95)
serieOsservata.ar.a = window(serieOsservata, start = c(1,7)) - r
ts.plot(serieOsservata.ar.a, serieOsservata.ar.pt$pred, col = c("black", "red"), 
xlim = c(30,50), lwd = 2)
```

#analisi residui

```
var(r)/var(window(serieOsservata, start = c(1, 1)))
s2_skewness(r)
s2_kurtosis(r)
layout(matrix(1:6, 2, 3, byrow = T))
plot(as.numeric(serieOsservata.ar$resid), pch = 20)
plot(serieOsservata.ar.a, serieOsservata.ar$resid, pch = 20)
acf(r, 28)
pacf(r, 28)
hist(serieOsservata.ar$resid, 20, freq = F)
lines(density(r), col = "blue")
lines(sort(r), dnorm(sort(r), mean(r), sd(r)), col = "red")
qqnorm(serieOsservata.ar$resid, pch = 20)
qqline(serieOsservata.ar$resid)
layout(1)
shapiro.test(serieOsservata.ar$resid)
```

```
ts.plot(serieOsservata.ar.a, serieOsservata.ar.pt$pred, col = c("black", "red"),
xlim = c(30,50), ylim = c(2,10), lwd = 2)

# non parametrico

lines(up, col = "blue", lwd = 2)

lines(lw, col = "blue", lwd = 2)

# parametrico

lines(serieOsservata.ar.pt$pred - serieOsservata.ar.pt$se, col = "green4", lwd =
2)

lines(serieOsservata.ar.pt$pred + serieOsservata.ar.pt$se, col = "green4", lwd =
2)
```

#confronto errori

```
nt = 18  # numero di test set

ft = 1  # unità di tempo nel futuro su cui valutare la previsione

n = length(serieOsservata)  # numero totale di anni

idt = start(serieOsservata)  # data di inizio della serie

fdt = end(serieOsservata)  # data di fine della serie

pdt = frequency(serieOsservata)  # periodo della seri

err_ar = rep(0, nt)

err_hw = rep(0, nt)

for (l in (n - nt - ft):(n - 1 - ft)) {

    # costruzione di train e test

    train = window(serieOsservata, idt, ts_data(idt, pdt, l))

    future = ts_data(idt, pdt, l + ft)

    test = window(serieOsservata, future, future)

    # HW

    train.hw = HoltWinters(train, alpha = 0.34, beta = 0.3, gamma = 0.5)

    err_hw[l - (n - nt - ft) + 1] = as.numeric(test) -
as.numeric(predict(train.hw,

        ft)[ft])

    # AR

    train.ar = ar(train, order.max = 14)

    err_ar[l - (n - nt - ft) + 1] = as.numeric(test) -
as.numeric(predict(train.ar,

        n.ahead = ft, se.fit = F)[ft])
```

```
}

sum(err_hw^2)/nt

sum(err_ar^2)/nt
```

#Minimi Quadrati

```
serieOsservata.ls = ar(serieOsservata, method = "ols")

serieOsservata.ls$order

ts.plot(serieOsservata, serieOsservata - serieOsservata.ls$resid, col =
c("black", "blue"))
```

#analisi residui

```
serieOsservata.ls.r = as.double(na.omit(serieOsservata.ls$resid))

serieOsservata.ls.a = as.double(na.omit(serieOsservata -
serieOsservata.ls$resid))

var(serieOsservata.ls.r)/var(window(serieOsservata, start = c(1, 1)))

s2_skewness(serieOsservata.ls.r)

s2_kurtosis(serieOsservata.ls.r)

layout(matrix(1:6, 2, 3, byrow = T))


plot(serieOsservata.ls.r, pch = 20)

plot(serieOsservata.ls.a, serieOsservata.ls.r, pch = 20)

acf(serieOsservata.ls.r, 28)

pacf(serieOsservata.ls.r, 28)

hist(serieOsservata.ls.r, 20, freq = F)

lines(density(serieOsservata.ls.r), col = "blue")

lines(sort(serieOsservata.ls.r), dnorm(sort(serieOsservata.ls.r),
mean(serieOsservata.ls.r), sd(serieOsservata.ls.r)), col = "red")

qqnorm(serieOsservata.ls.r, pch = 20)

qqline(serieOsservata.ls.r)

layout(1)

shapiro.test(serieOsservata.ls.r)

serieOsservata.hw = HoltWinters(serieOsservata, seasonal = "m")

ts.plot(serieOsservata, serieOsservata - serieOsservata.ls$resid,
serieOsservata.hw$fitted[, 1], col = c("black", "red",

    "blue"), xlim = c(30,50), ylim = c(2,10),lwd = 2)

serieOsservata.hw.pt = predict(serieOsservata.hw, 12)
```

```
ts.plot(serieOsservata, serieOsservata.ls.pt$pred, serieOsservata.hw.pt, col =
c("black", "red","blue"), xlim = c(30,50), ylim = c(2,10),lwd = 2)

lines(serieOsservata.ls.pt$pred - serieOsservata.ls.pt$se, col = "green4", lwd =
2)

lines(serieOsservata.ls.pt$pred + serieOsservata.ls.pt$se, col = "green4", lwd =
2)
```

#confronto errori

```
nt = 20  # numero di test set

ft = 1  # unità di tempo nel futuro su cui valutare la previsione

n = length(serieOsservata)  # numero totale di anni

idt = start(serieOsservata)  # data di inizio della serie

fdt = end(serieOsservata)  # data di fine della serie

pdt = frequency(serieOsservata)  # periodo della seri

err_ls = rep(0, nt)

err_hw = rep(0, nt)

for (l in (n - nt - ft):(n - 1 - ft)) {

    # costruzione di train e test

    train = window(serieOsservata, idt, ts_data(idt, pdt, l))

    future = ts_data(idt, pdt, l + ft)

    test = window(serieOsservata, future, future)

    # HW

    train.hw = HoltWinters(train, alpha = 0.1, beta = 0.92, gamma = 0.36,

        seasonal = "multiplicative")

    err_hw[l - (n - nt - ft) + 1] = as.numeric(test) -
as.numeric(predict(train.hw,

        ft)[ft])

    # AR

    train.ls = ar(train, order.max = 21, method = "ols")

    err_ls[l - (n - nt - ft) + 1] = as.numeric(test) -
as.numeric(predict(train.ls,

        n.ahead = ft, se.fit = F)[ft])

}

sum(err_hw^2)/nt

sum(err_ls^2)/nt
```