1. **SET con PARAMETRI CONSERVATIVI**

Abbiamo visto che un modo per lisciare il trend e proseguirlo bene nel futuro consiste nel prendere il trend di stl (basso k) ed applicargli SET. Invece, l’applicazione diretta di SET a volte è più difettosa (trend oscillante, predizione troppo condizionata dagli ultimi valori).

Però, si può provare a mano assegnando valori conservativi a SET.

-----inizio recupero dati-----

X = scan("clipboard")

11849 1316 4712 800 5097 3270 5390 2135 5962 5795

9271 6864 4247 7961 7191 4970 5012 2929 7363 4907

4700 8219 8674 8263 4294 6097 9115 8924 12561 8626

9559 1706 7405 8057 6463 7595 6702 11052 8422 10019

9594 6443 12052 3535 7962 12876 10614 6469 9396 8421

10895 16583 13724 15362 10740 6999 8168 11241 12206 9600

13865 11190 11306 9760 16678 17245 17059 6331 17635 25429

24124 8025 16168 23902 15618 24579 23023 16311 18848 8986

19745 21513 17378 17839 20271 13917 22520 30279 33373 27728

29400 12759 32293 39832 21975 13304 21511 24382 27790 19053

30140 18024 39378 9351 29202 30944 30984 51181 25809 23035

33939 26385 37051 27823 26570 27587 23434 40944 30355 37954

21883 28773 37127 41490 40786 48097 42233 31002 44818 52006

44235 47597 41311 29891 65657 46764 56831 45358 42840 22724

54484 60902 59253 30692 54913 49463 61164 53140 53932 61780

76395 25181 52878 59362 36356 22600 69768 62622 48934 43957

46525 47652 68435 23165 65990 60518 38106 21206

-----fine recupero dati-----

-----inizio metodi già visti-----

SET = HoltWinters(X, gamma=FALSE)

plot(SET, predict(SET,12))

accessori.auto = ts(X, frequency=12, start=c(1995,1))

trend = stl(accessori.auto,3)$time.series[,2]

plot(trend)

par(mfrow=c(1,2))

plot(SET, predict(SET,12))

plot(trend)

-----fine metodi già visti-----

Ecco SET con parametri conservativi:

SET.cons = HoltWinters(X, alpha=0.1, beta=0.1, gamma=FALSE)

plot(SET.cons, predict(SET.cons,12))

plot(SET, predict(SET,12))

Il risultato in fase di predizione è buono. Sono senza senso i primi valori. E’ colpa dell’inizializzazione di default di R: pendenza iniziale data da quella dei primi due valori della serie. Basta cambiare:

SET.cons = HoltWinters(X, alpha=0.1, beta=0.1, gamma=FALSE, b.start=0)

plot(SET.cons, predict(SET.cons,12))

plot(SET, predict(SET,12))

1. **ALTRO ESEMPIO SU METODI REGRESSIVI PER LE SERIE STORICHE**

Sviluppiamo l’esempio difficile anche con un modello AR. Anche per vedere una variante dei comandi.

> length(X)

[1] 168

y=X[13:168]; x1=X[12:167]; x12=X[1:156]

Reg = lm(y~x1+x12)

P=1:180; P[1:168]=X

for (i in 1:12) {

P[168+i]=coef(Reg)%\*%c(1,P[168+i-1],P[168+i-12])

}

Pplus = ts(P, frequency =12, start=c(1995,1))

Si noti l’uso di coef(Reg)%\*%c(…).

Si noti inoltre la somiglianza col risultato di HW, che ad es. può costituire una conferma della bontà di entrambi:

HW = HoltWinters(accessori.auto)

par(mfrow=c(1,2))

plot(HW,predict(HW,12))

plot(Pplus)

par(mfrow=c(1,2))

plot(predict(HW,12))

ts.plot(Pplus[169:180])

Un vantaggio del metodo auto regressivo, è il seguente.

Se ci fossero delle altre, pur deboli, periodicità, il metodo ne potrebbe tener conto. Da

ts.plot(X[100:168]) ; acf(X[100:168])

si vede che c’è un lieve ritardo a 4 mesi (qualche forma di gestione quadrimestrale delle vendite?). Lo possiamo introdurre nel modello:

y=X[13:168]; x1=X[12:167]; x4=X[9:164]; x12=X[1:156]

Reg = lm(y~x1+x4+x12)

P=1:180; P[1:168]=X

for (i in 1:12) {

P[168+i]=coef(Reg)%\*%c(1,P[168+i-1],P[168+i-4],P[168+i-12])

}

Pplus2 = ts(P, frequency =12, start=c(1995,1))

Possiamo vedere i risultati a confronto:

par(mfrow=c(1,2))

plot(Pplus2)

plot(Pplus)

par(mfrow=c(1,2))

ts.plot(Pplus2[169:180])

ts.plot(Pplus[169:180])

Non abbiamo strumenti precisi per giudicare quale sia meglio. Possiamo però esaminare l’importanza dei fattori nella regressione:

summary(Reg)