1. **GRAFICO DEI DUE R^2**

Si riprenda l’esercitazione del 19/10, relativa alla tabella IB

IB <- read.table ("clipboard", header=TRUE)

PLIC SC SA.SC TD TMI

Piem 0.088 0.471 -0.707 -0.607 -0.3950

Vaos -1.545 0.348 -0.642 -0.813 1.5780

Lomb 0.202 1.397 -0.836 -0.790 -0.5380

TrAA 0.677 0.435 -1.269 -0.966 -0.0750

Vene 0.088 1.334 -1.210 -0.848 -0.4970

FrVG 0.639 -0.005 -1.028 -0.804 -1.3010

Ligu 1.190 -0.247 0.470 -0.429 -0.3540

EmRo 0.658 1.177 -1.315 -0.863 -0.3470

Tosc 0.126 1.092 -0.795 -0.644 -1.3550

Umbr -1.431 0.675 -0.140 -0.524 -1.2870

Marc 0.278 1.090 -0.265 -0.702 -0.0006

Lazi 2.329 0.546 -0.080 -0.113 -0.0140

Abru 0.335 -0.373 0.402 -0.456 0.0400

Moli 0.658 -1.289 0.065 0.451 -1.1510

Camp -1.811 -1.314 2.031 1.664 0.4140

Pugl -0.766 -0.926 1.038 0.648 1.1090

Basi -0.747 -1.154 0.661 0.844 2.0010

Cala -0.500 -1.727 1.571 2.153 0.6320

Sici -0.918 -1.130 1.332 1.517 1.7830

Sard 0.449 -0.403 0.717 1.285 -0.2380

Ricordiamo i risultati qui rilevanti:

fit.1 = lm(TD~PLIC+SC+SA.SC+TMI, data=IB)

Multiple R-squared: 0.8464, Adjusted R-squared: 0.8055

F-statistic: 20.67 on 4 and 15 DF, p-value: 5.801e-06

[Curiosità: fit.1 = lm(IB$TD~IB$PLIC+IB$SC+IB$SA.SC+IB$TMI) fornisce lo stesso risultato.]

fit.2 = lm(TD~SC+SA.SC+TMI, data=IB)

Multiple R-squared: 0.8464, Adjusted R-squared: 0.8176

F-statistic: 29.4 on 3 and 16 DF, p-value: 9.597e-07

fit.3 = lm(TD~SC+SA.SC, data = IB)

Multiple R-squared: 0.8464, Adjusted R-squared: 0.8283

F-statistic: 46.83 on 2 and 17 DF, p-value: 1.216e-07

fit.4 = lm(TD~SC+SA.SC+0, data=IB)

Multiple R-squared: 0.8464, Adjusted R-squared: 0.8293

F-statistic: 49.58 on 2 and 18 DF, p-value: 4.766e-08

fit.5 = lm(TD~SA.SC+0, data=IB)

Multiple R-squared: 0.8198, Adjusted R-squared: 0.8098

F-statistic: 81.88 on 1 and 18 DF, p-value: 4.064e-08

Riassumiamo i risultati di due degli indicatori principali nella seguente tabella (è stata compilata manualmente; sarebbe interessante automatizzare l’uso di tali numeri):

fit.1: Multiple R-squared: 0.8464, Adjusted R-squared: 0.8055

fit.2: Multiple R-squared: 0.8464, Adjusted R-squared: 0.8176

fit.3: Multiple R-squared: 0.8464, Adjusted R-squared: 0.8283

fit.4: Multiple R-squared: 0.8464, Adjusted R-squared: 0.8293

fit.5: Multiple R-squared: 0.8198, Adjusted R-squared: 0.8098

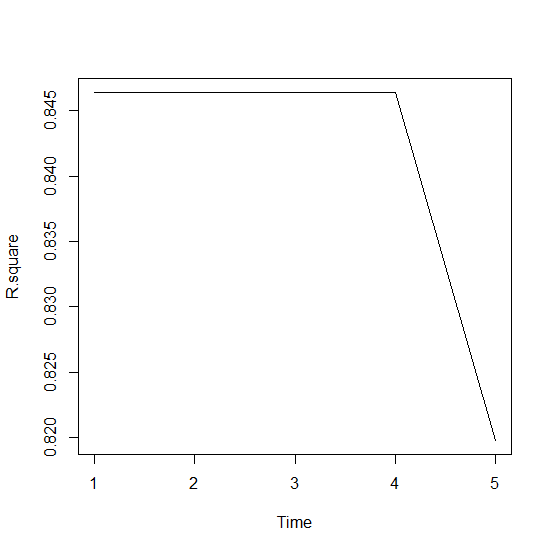
Si vede che fino a fit.4 c’è sicuramente un miglioramento, mentre fit.5 è peggiore da questi punti di vista.

Visualizziamo come grafici questi risultati:

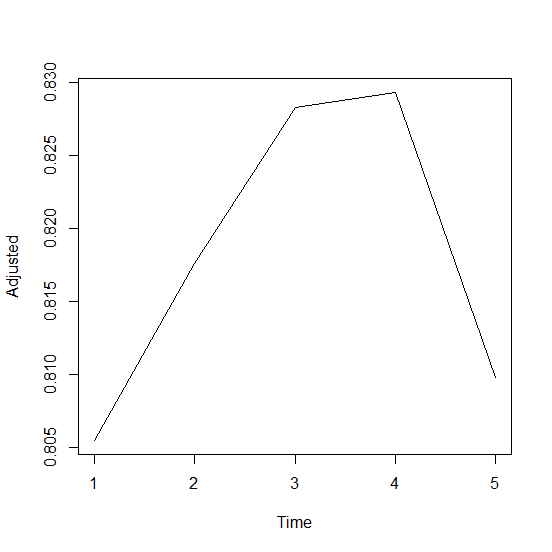
R.square = c(0.8464, 0.8464, 0.8464, 0.8464, 0.8198)

Adjusted = c(0.8055, 0.8176, 0.8283, 0.8293, 0.8098)

ts.plot(R.square)



ts.plot(Adjusted)



Multiple R-squared ha un “gomito” passando da 2 ad 1 fattore

Adjusted R-squared ha un massimo per 2 fattori.

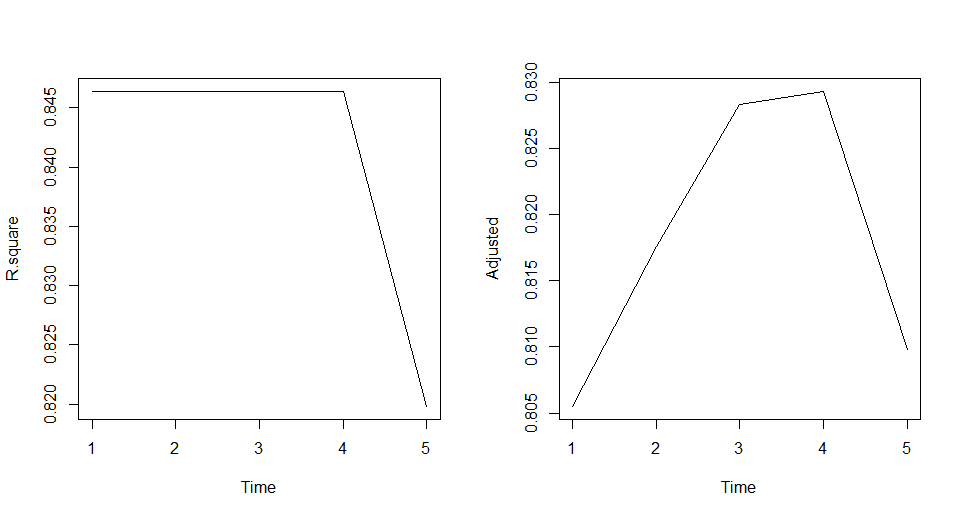
Entrambi quindi indicano che 2 fattori è il punto dove fermarsi.

Possiamo mettere i due grafici su una stessa schermata:

par(mfrow=c(1,2))

ts.plot(R.square)

ts.plot(Adjusted)



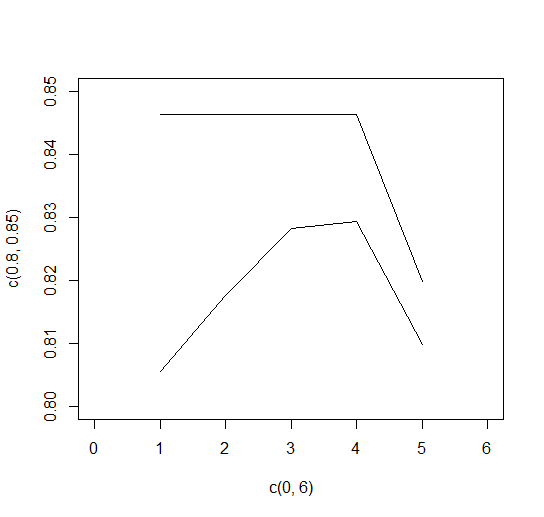
oppure metterli su uno stesso grafico, ma bisogna lavorare su alcuni parametri:

par(mfrow=c(1,1))

plot(c(0,6),c(0.8,0.85),type="n")

lines(R.square)

lines(Adjusted)



**Esercizio. Costruire artificialmente una tabella con 5 colonne e 30 righe, in cui le prime due colonne siano molto allineate e l’ultima colonna sia molto correlata ad esse. Applicare poi la regressione lineare multipla, in cui l’ultima colonna sia l’output del modello ed osservare che, nonostante il forte legame, i p-value sono scarsi.**

1. **PREVISIONE CON MODELLO DI REGRESSIONE**

Illustriamo ora come si usa un modello di regressione per fare previsioni.

*Nota: la previsione si fa per diverse ragioni: una è prevedere un valore incognito; l’altra, anche se il valore è noto, può essere di valutare come tale valore si colloca rispetto al modello*.

Supponiamo di avere solo la tabella di alcune regioni italiane. Ad esempio, togliamo FRVG e Basi dalla precedente (fingiamo di non averle):

IB.1 = IB[-6,]

IB.2=IB.1[-16,]

Calcolare cor(IB.2) e notare che certe correlazioni sono calate.

Costruiamo il modello per TD plausibilmente più interessante:

fit = lm(TD~ SC+ SA.SC+0,data=IB.2)

Osservare summary(fit) per accertarsi che continua ad essere un buon modello.

Passiamo alla previsione. Supponiamo di avere i valori di SC.SA ed SC di FRVG e Basi:

C <- read.table ('clipboard', header=TRUE)

PLIC SC SA.SC TMI

FrVG 0.639 -0.005 -1.028 -1.3010

Basi -0.747 -1.154 0.661 2.0010

“Prevediamo” il valore di TD usando il modello e questi valori:

TD.FrVG.prev = fit$coef %\*%c(-0.005, -1.028)

oppure

TD.FrVG.prev = fit$coef %\*%c(C[1,2],C[1,3])

TD.Basi.prev = fit$coef %\*% c(C[2,2],C[2,3])

> c(TD.FrVG.prev, TD.Basi.prev)

[1] -0.6396872 0.7831046

Confrontati coi valori veri:

TD

FrVG -0.804

Basi 0.844

Il risultato non è così cattivo. Coglie il “segno” rispetto alla media nazionale ed anche l’ordine di grandezza della variazione.

Nei termini accennati sopra secondo cui la previsione si può fare per valutare come si colloca la regione rispetto allo schema nazionale, possiamo dire che FrVG va meglio di quanto predirebbe il modello nazionale, Basi va peggio.

Proviamo col modello completo, con tutti i fattori:

fit.1 = lm(TD~PLIC+SC+SA.SC+TMI, data=IB.2)

TD.FrVG.prev = fit.1$coef %\*% c(1,C[1,1],C[1,2],C[1,3],C[1,4])

TD.Basi.prev = fit.1$coef %\*% c(1,C[2,1],C[2,2],C[2,3],C[2,4])

> c(TD.FrVG.prev, TD.Basi.prev)

[1] -0.6197502 0.7701555

Il risultato è ugualmente buono, quindi in questo caso l’eliminazione non ha prodotto grossi vantaggi. Si noti tuttavia che, pur di poco, il risultato è migliore con meno fattori. Gli altri, invece che contribuire ad un fit più preciso, fungono da disturbo.

Considerazione conclusiva. Il metodo ora esposto di eliminare uno o pochi individui, trovare il modello relativo a quelli mantenuti e testarlo sugli individui eliminati viene detto “cross-validation” e può essere usato per testare la bontà di un modello rispetto ad un altro (confrontare modelli). Ovviamente R^2 ecc. sono già indicatori che servono a confrontare modelli; ad essi si aggiunge la cross-validation.