STATISTICA 1, metodi matematici e statistici

Introduzione al linguaggio R

Esercitazione 8: 27-05-2004

Luca Monno

Università degli studi di Pavia

luca.monno@unipv.it
http://www.lucamonno.it

Regressione con polinomi

Nella lezione precedente abbiamo utilizzato un modello di regressione lineare per studiare la percorrenza in funzione della cilindrata. Il modello però non si adattava bene ai dati.
Proviamo ad utilizzare invece di una retta un polinomio:

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \ldots + \beta_p x^p + \epsilon$$

Scegliamo ad esempio $p = 3$ e stimiamo i parametri del modello, ricaricando prima il dataset:

```
> auto = read.table("auto.dat", header = T)
> attach(auto)
> y = percorr.urbana
> x = cilindrata
```

```
> formula = y ~ x
> formula1 = y ~ x + I(x^2) + I(x^3)
```

La funzione `I` serve ad indicare che non stiamo facendo una somma algebrica ma stiamo solamente aggiungendo delle variabile del modello.

```
> mod = lm(formula)
> mod1 = lm(formula1)
> summary(mod)
> summary(mod1)
```

Il valore R^2 è aumentato ma la componente cubica ha un p-value molto alto quindi accettiamo l'ipotesi che $\beta_3 = 0$ quindi x^3 non èmolto significativa per il modello:

```
> formula2 = y ~ x + I(x^2)
> mod2 = lm(formula2)
```

Notiamo che anche togliendo la componente x^3 il valore R^2 è rimasto praticamente invariato e ora la seconda componente ha un p-value praticamente nullo.
Graficamente il modello può essere rappresentato disegnando sopra al diagramma di dispersione la parabola e la cubica stimata.

```
> plot(y ~ x)
> beta2 = mod2$coef
> beta3 = mod1$coef
> xx = seq(1, 5.5, length = 200)
> lines(xx, beta2[1] + beta2[2] * xx + beta2[3] * xx^2, col = 4
> lines(xx, beta3[1] + beta3[2] * xx + beta3[3] * xx^2 + beta3[
+     xx^3, col = 5)
```

Ci sono altre diagnostiche importanti per vedere se il modello si adatta bene ai dati, in particolare è molto importante vedere se i residui soddisfano l'ipotesi di omoschedasticità. Se passiamo come argomento al comando `plot` un modello, R costruisce immediatamente alcune diagnostiche grafiche:

```
> par(mfrow = c(2, 2))
> plot(mod1)
> par(mfrow = c(1, 1))
```

Nel primo grafico si mettono a confronto i residui con i valori stimati, il secondo è un qqplot per vedere i residui hanno distribuzione normale. Il quarto evidenzia i dati che hanno molta influenza nelle stima del modello.
Osserviamo che la varibilità dei residui non è costante, quindi è presente eteroschedasticità. Questo può derivare da una possibile etereogeneità del campione.

Se osserviamo le caratteristiche delle macchine presenti nel dataset ci accorgiamo che, oltre alla cilindrata, ci sono molte altre variabili che dovrebbero influenzare la percorrenza: per esempio l'alimentazione. quindi

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 I_A + \epsilon$$

dove I_A è una funzione che vale 1 se la macchina è diesel e 0 se a benzina.

```
> a = alimentazione
> formula4 = y ~ x + I(x^2) + I(x^3) + a
> mod4 = lm(formula4)
```

Attraverso il comando `summary` che il valore R^2 è molto aumentato e la componente alimentazione è significativa. Graficamente la situazione migliora:

```
> plot(y ~ x, type = "n")
> d = (a == "benz")
> points(y[d] ~ x[d], col = 2, pch = 2)
> points(y[!d] ~ x[!d], col = 3, pch = 3)
```

Notiamo che le macchine diesel denotate da punti verdi si trovanno sempre sopra le macchine a benzina. Disegnamo le curve del modello:

```
> beta4 = mod4$coef
> lines(xx, beta4[1] + beta4[2] * xx + beta4[3] * xx^2 + beta4[
+     xx^3, col = 2)
> lines(xx, beta4[1] + beta4[2] * xx + beta4[3] * xx^2 + beta4[
+     xx^3 + beta4[5], col = 3)
```

ma le diagnostiche grafiche sui residui non sono ancora buone:

```
> par(mfrow = c(2, 2))
> plot(mod4)
> par(mfrow = c(1, 1))
```

Trasformazioni di variabili

Oltre ai polinomi, sono ultilizzate numerose altre trasformazioni, per esempio quella che permette di passare in scala logartimica:

$$\log(y) = \beta_0 + \beta_1 \log(x) + \epsilon$$

Oltre alla percorrenza consideriamo anche l'alimentazione e il peso:

```
> p = peso
> formula5 = log(y) ~ log(x) + a + log(p)
> mod5 = lm(formula5)
```

Il valore R^2 restituito è molto alto che indica un ottimo adattamento del modello ai dati.
Attenzione! Questo valore trovato non è confrontabile con gli altri perché siamo in scala logartmica.
Anche le diagnostiche grafiche sono soddisfacenti:

```
> par(mfrow = c(2, 2))
> plot(mod5)
```