Laboratorio 4

Analisi dei residui

4.1 Analisi dei dati CEMENT.DAT

I dati riportati nel file `cement.dat` si riferiscono ad uno studio sulla resistenza del cemento alla tensione. La resistenza dipende, tra le altre cose, dal tempo di essicazione. Nello studio si è misurata la resistenza alla tensione di lotti di cemento sottoposti a diversi tempi di essicazione. Si studi la relazione tra la resistenza alla tensione e il tempo di essicazione.
In questo caso il tempo è la variabile esplicativa e la resistenza è la variabile risposta.

```
> cement <- read.table("I:/modelli/cement.dat",
                       col.names=c("tempo", "resist"))
> attach(cement)
```

Proviamo una prima analisi esplorativa dei dati:

```
> plot(resist ~ tempo)
```

Il grafico indica chiaramente una relazione non lineare. Un modello del tipo $\texttt{resist} = \beta_1 + \beta_2 \texttt{tempo} + \varepsilon$ non sembra appropriato. Possiamo allora cercare qualche trasformazione delle variabili che ci riporti ad una relazione più lineare.
Generalmente, si preferisce trasformare le variabili esplicative. Proviamo allora a trasformare la variabile `tempo`. Si noti l'utilizzo della funzione `par` con l'opzione `mfrow`. Questo permette di visualizzare in un un'unica finestra $2 \times 2 = 4$ grafici.

```
> par(mfrow=c(2,2))
> plot(log(tempo), resist)
> plot(1/(tempo), resist)
> plot(1/sqrt(tempo), resist)
> plot(sqrt(tempo), resist)
> par(mfrow=c(1,1))
```

Le prime tre trasformazioni pare linearizzino in maniera soddisfacente la relazione, in particolare la terza. Adottiamo quindi la trasformazione

```
> x <- 1/sqrt(tempo)
```

e procediamo specificando il modello di regressione

$$\texttt{resist} = \beta_1 + \frac{\beta_2}{\texttt{tempo}} + \varepsilon$$

A questo punto utilizziamo una nuova funzione `lm`. Questa fornisce le stime di massima verosimiglianza per un modello lineare quando gli errori hanno distribuzione normale.

```
> fit <- lm( resist ~ x )
```

Notiamo la sintassi di `resist~x` . A sinistra di ~ vi è il regressore, a destra la variabile esplicativa. L'intercetta β_1 è automaticamente inclusa. Abbiamo creato un oggetto, che abbiamo chiamato `fit`, di tipo `lm`. Un oggetto è qualcosa di più complicato di un vettore o di una matrice. È una lista di elementi su cui si può applicare una serie di funzioni. Ad esempio con il comando seguente vediamo i risultati dell'adattamento.

```
> summary(fit)

Call:
lm(formula = resist ~ x)

Residuals:
     Min       1Q   Median       3Q      Max
-3.79469 -1.25666 -0.05666  1.89544  3.10531

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   45.655      1.023   44.63  < 2e-16 ***
x            -32.599      1.764  -18.48 1.34e-13 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.133 on 19 degrees of freedom
Multiple R-Squared: 0.9473,     Adjusted R-squared: 0.9445
F-statistic: 341.4 on 1 and 19 DF,  p-value: 1.337e-13
```

Come si può osservare, otteniamo diverse statistiche. Notiamo che entrambi i coefficienti sono fortemente significativi e così pure (come è ovvio) il test F sulla bontà dell'adattamento complessivo del modello.
L'oggetto `fit` contiene più quantità

```
> names(fit)
 [1] "coefficients"  "residuals"     "effects"       "rank"
 [5] "fitted.values" "assign"        "qr"            "df.residual"
 [9] "xlevels"       "call"          "terms"         "model"
```

Proviamo a considerare i residui e i valori previsti dal modello.

```
> res <- resid(fit)
> fit.val <- fitted(fit)
```

I residui contenuti in `fit` e che abbiamo appena salvato nel vettore `res` sono le quantità $e_i = y_i - \hat{y}_i$. Per ottenere i residui standardizzati, $e_i^* = e_i/(s\sqrt{1-h_i})$, dobbiamo utilizzare la funzione `rstandard`

```
> res.standard <- rstandard(fit)
```

Alcuni grafici che possiamo fare per verificare la linearità della relazione, l'omoschedasticità degli errori e l'indipendenza degli errori sono:

1. il grafico (i, e_i^*), utile soprattutto se le osservazioni sono in ordine temporale;
2. il grafico $(\hat{y}_i, e_i^*)$;
3. il grafico $(\hat{y}_i, y_i)$;
4. il grafico (x_i, e_i^*).

Il grafico (2), dei residui rispetto ai valori stimati, mostra una maggiore variabilità dei residui per valori stimati elevati. Questo sembrerebbe indicare che la varianza non è costante, ossia che i residui non sono omoschedastici.

```
> plot(fit.val, res.standard)
```

Lo stesso andamento è mostrato anche dal grafico (3) dei valori osservati sui valori stimati, anche se in modo meno evidente.

```
> plot(fit.val, resist)
```

Per quanto riguarda la normalità dei residui, appaiono lievi deviazioni sulla coda destra.

```
> par(pty='s')
> qqnorm(res.standard, xlim=c(-2,2), ylim=c(-2,2))
> qqline(res.standard)
```

Possiamo provare a vederlo anche con i soliti strumenti:

```
> hist(res.standard, freq=F)
> lines(density(res.standard))
> boxplot(res.standard)
```

Complessivamente, la normalità appare soddisfacente considerando la bassa numerosità del campione. Concludendo, il modello interpola i dati abbastanza bene; esso risulta peraltro un po' carente per quanto riguarda la omoschedasticità del termine di errore.
E adesso terminiamo.

```
> detach(cement)
```

Esercizio: Produrre il grafico (i, e_i^*) e commentarlo. Spesso, in pratica, anzichè utilizzare i residui standardizzati si utilizzano i residui non standardizzati e_i. Ripetere le analisi grafiche precedenti con i residui non standardizzati e commentare eventuali somiglianze o differenze.

4.2 Analisi dei dati WINDMILL.DAT

Un ingegnere sta provando una turbina eolica per generare corrente elettrica. Sono disponibili un certo numero di osservazioni sulla corrente generata e sulla corrispondente velocità del vento e si è interessati alla relazione che intercorre tra velocità del vento e corrente generata. I dati sono contenuti nel file `windmill.dat`.

```
> windmill <- read.table("I:/modelli/windmill.dat", header=T)
> windmill
    wind    dc
1   5.00 1.582
2   6.00 1.822
3   3.40 1.057
4   2.70 0.500
    . . .
> attach(windmill)
```

Per esplorare graficamente la relazione esistente tra velocità del vento e corrente generata facciamo un diagramma di dispersione.

```
> plot(wind, dc)
```

Il grafico mostra una evidente relazione tra le due variabili. Proviamo dapprima ad ipotizzare un legame lineare.

```
> fit <- lm( dc ~ wind )
> summary(fit)

Call:
lm(formula = dc ~ wind)

Residuals:
     Min       1Q   Median       3Q      Max
-0.59869 -0.14099  0.06059  0.17262  0.32184

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.13088    0.12599   1.039     0.31
wind         0.24115    0.01905  12.659 7.55e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2361 on 23 degrees of freedom
Multiple R-Squared: 0.8745,     Adjusted R-squared: 0.869
F-statistic: 160.3 on 1 and 23 DF,  p-value: 7.546e-12
```

Per verificare la bontà del modello, possiamo valutare la significatività dei coefficienti e passare poi all'analisi dei residui.

Appare evidente che entrambi i coefficienti sono fortemente significativi. Dato il risultato osservato sul coefficiente angolare, non sorprende il risultato del test F per la bontà complessiva del modello:

```
> 12.659^2
[1] 160.2503
```

Passiamo ora all'analisi dei residui.

```
> res <- rstandard(fit)
> fit.val <- fitted(fit)
> plot(fit.val, res)
> plot(wind, res)
```

I grafici dei residui rispetto ai valori adattati e rispetto alla variabile esplicativa indicano un chiaro andamento parabolico dei residui standardizzati (lo stesso si evincerebbe utilizzando i residui e_i). In particolare, il grafico (x_i, e_i^*) dice che in corrispondenza di valori bassi e alti della velocità del vento, il modello sistematicamente sottostima il valore della della corrente generata, mentre in corrispondenza di valori centrali della velocità, il modello sovrastima la corrente generata. Questo andamento suggerisce che il modello non coglie in maniera appropriata la dipendenza della variabile risposta dalla esplicativa.

```
> qqnorm(res)
```

Il grafico quantile–quantile mostra qualche scostamento dalla normalità per i residui di segno positivo. Da questo potremmo desumere che la distribuzione dei residui sia asimmetrica. In conclusione, l'analisi dei residui non appare soddisfacente, nonostante i risultati ottenuti nei test di significatività.
Come possiamo rimediare? Il grafico (x_i, e_i^*) mostra una relazione di tipo quadratico. Quindi il modello potrebbe essere migliorato introducendo un ulteriore regressore ovvero la velocità del vento al quadrato. Potremmo quindi passare da una regressione semplice ad una regressione multipla. Tuttavia, il diagramma di dispersione

```
> plot(wind, res)
```

mostra che la relazione tra corrente e velocità è monotona decrescente. Proviamo allora, invece di una regressione multipla con un termine quadratico, a considerare un modello del tipo

$$\texttt{corrente} = \beta_1 + \frac{\beta_2}{\texttt{velocità}} + \varepsilon$$

```
> fit.inv <- lm( dc ~ I(1/wind) )
> summary(fit.inv)

Call:
lm(formula = dc ~ I(1/wind))

Residuals:
```

```
     Min       1Q   Median       3Q      Max
-0.20547 -0.04941  0.01100  0.08352  0.12204

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   2.9789     0.0449   66.34   <2e-16 ***
I(1/wind)    -6.9345     0.2064  -33.59   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.09417 on 23 degrees of freedom
Multiple R-Squared:  0.98,      Adjusted R-squared: 0.9792
F-statistic:  1128 on 1 and 23 DF,  p-value:     0
```

Consideriamo dei valori della velocità del vento al di fuori del *range* di variazione dei valori osservati.

```
> new.wind <- 1:30
> beta1 <- coef(fit.inv)[1]
> beta2 <- coef(fit.inv)[2]
> fit.val <- beta1  +beta2/new.wind
```

e rappresentiamoli

```
> plot(wind, dc, xlim=c(0,30), ylim=c(0,4))
> lines(1:30, fit.val)
```

L'adattamento misurato da R^2 è migliorato. I test rilevano significatività dei coefficienti e la bontà complessiva del modello. Inoltre, i risultati delle analisi dei residui appaiono migliori. In particolare consideriamo

```
> res.inv <- rstandard(fit.inv)
> fitted.inv <- fitted(fit.inv)
> plot(wind, res.inv)
> plot(fitted.inv, res.inv)
```

Esercizio: Completare l'analisi dei residui.