Laboratorio 7

Ancora sulla regressione multipla

7.1 Analisi del dataset HILLS.DAT

I dati contenuti nel file `hills.dat` riguardano il record registrato in 35 corse campestri effettuate sulle montagne scozzesi, la distanza coperta nelle corse e il dislivello affrontato. Si vuole costruire un modello che metta in relazione il tempo di record con la distanza ed il dislivello.

```
> hills<-read.table("I:\\modelli\\hills.dat")
> attach(hills)
```

Procediamo all'analisi preliminare dei dati:

```
> pairs(hills)
```

Dall'analisi, si nota una relazione di tipo crescente tra la variabile risposta (`time`) ed entrambe le esplicative (`dist`, `climb`). La relazione appare particolarmente evidente se si considera la distanza. Proviamo a calcolare le correlazioni semplici tra le variabili.

```
> cor(hills)
```

L'analisi dei coefficienti di correlazione conferma l'intuizione. Notiamo anche una certa correlazione tra le variabili esplicative.
Procediamo quindi a costruire un modello di regressione lineare includendo prima la distanza e successivamente, se dovesse risultare utile, il dislivello.

```
> fit <- lm(time~dist)

> summary (fit)

Call:
```

```
lm(formula = time ~ dist)

Residuals:
    Min      1Q  Median      3Q     Max
-35.745  -9.037  -4.201   2.849  76.170

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -4.8407     5.7562  -0.841    0.406
dist          8.3305     0.6196  13.446    6e-15 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 19.96 on 33 degrees of freedom
Multiple R-Squared: 0.8456,     Adjusted R-squared: 0.841
F-statistic: 180.8 on 1 and 33 degrees of freedom,      p-value: 6.106e-015
```

Il modello più semplice mostra una forte significatività della esplicativa (si vedano i test opportuni) ed una non significatività dell'intercetta, come ci si poteva attendere tenendo conto del significato delle variabili. Il coefficiente $R^2 = \sum(\hat{y}_i - \bar{y})^2 / \sum(y_i - \bar{y})^2$ ci dice che circa l'85% della variabilità della risposta è dovuto al dipendere lineare del tempo di record dalla distanza. Il rimanente 15% è dovuto all'intervento di variabili esplicative non ancora considerate (come il dislivello, nel nostro esempio) ed all'errore.
Può il dislivello fornire ulteriori spiegazioni della quota rimanente di variabilità? Proviamo a vedere.

```
> res <- rstandard(fit)
> plot(climb, res)
> identify(climb, res, dimnames(hills)[[1]])
```

In effetti il grafico evidenzia una forte relazione tra i residui e la variabile dislivello, a significare che il modello può ancora essere migliorato tramite l'inserimento del dislivello. Prima però eliminiamo l'intercetta.

```
> fit1 <- lm(time~dist-1)
> summary(fit1)

Call:
lm(formula = time ~ dist - 1)

Residuals:
     Min       1Q   Median       3Q      Max
-28.7646 -11.0285  -6.8327   0.5607  78.0847

Coefficients:
     Estimate Std. Error t value Pr(>|t|)
```

```
dist   7.9083     0.3615   21.88   <2e-16 ***
---
Signif. codes:  0  ‘***’  0.001  ‘**’  0.01  ‘*’  0.05  ‘.’  0.1  ‘ ’  1

Residual standard error: 19.87 on 34 degrees of freedom
Multiple R-Squared: 0.9337,     Adjusted R-squared: 0.9317
F-statistic: 478.6 on 1 and 34 degrees of freedom,      p-value:     0
```

Si noti che il valore di R^2 aumenta!! Ciò non è dovuto al fatto che il modello è migliorato a seguito dell'eliminazione dell'intercetta, ma al fatto che R^2 per un modello senza intercetta è calcolato come $\sum \hat{y}_i^2 / \sum y_i^2$. Ciò rende il valore di R^2 ottenuto per `fit` non confrontabile con quello ottenuto per `fit1` (in altre parole, è errato affermare che *il modello è migliorato perchè R^2 è aumentato*).
Inseriamo ora `climb`.

```
> fit2 <- update(fit1, .~.+climb)

> summary(fit2)

Call:
lm(formula = time ~ dist + climb - 1)

Residuals:
    Min      1Q  Median      3Q     Max
-18.089 -10.053  -5.539  -3.180  58.235

Coefficients:
      Estimate Std. Error t value Pr(>|t|)
dist  5.605651   0.551046  10.173 1.05e-11 ***
climb 0.010280   0.002118   4.853 2.84e-05 ***
---
Signif. codes:  0  ‘***’  0.001  ‘**’  0.01  ‘*’  0.05  ‘.’  0.1  ‘ ’  1

Residual standard error: 15.41 on 33 degrees of freedom
Multiple R-Squared: 0.9613,     Adjusted R-squared: 0.959
F-statistic: 409.8 on 2 and 33 degrees of freedom,      p-value:     0
```

Anche il dislivello è fortemente significativo e tutti i test rilevano la bontà del modello. Il miglioramento (espresso in termini di devianza residua) ottenuto mediante l'inserimento di `climb` è valutabile utilizzando la funzione `anova`, che fornisce il test per verificare l'ipotesi nulla, che valga il modello ridotto senza la variabile esplicativa `climb`, contro l'alternativa, che valga il modello completo che include sia `dist` che `climb`.

```
> anova(fit1,fit2)
```

```
Analysis of Variance Table

Model 1: time ~ dist - 1
Model 2: time ~ dist + climb - 1
  Res.Df Res.Sum Sq Df  Sum Sq F value   Pr(>F)
1     34    13423.2
2     33     7832.5  1  5590.7  23.555 2.84e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

La statistica F esprime la riduzione relativa di devianza residua dovuta all'inserimento di `climb`. La statistica è ottenuta come:

$$F = \frac{(13423.2 - 7832.5)/(34 - 33)}{7832.5/33}$$

ed è sotto l'ipotesi nulla, realizzazione di una $F_{1,33}$. Se il test è significativo, l'inserimento della variabile ha portato ad una riduzione statisticamente significativa della devianza residua del modello.
Passiamo all'analisi dei residui.

```
> par(mfrow=c(2,2))
> plot(fit2)
```

L'analisi dei residui mostra delle particolarità: alcuni valori (11,7,18) paiono anomali.

```
> par(mfrow=c(1,2))
> res <- rstandard(fit2)
> plot(time, fitted(fit2))
> abline(0,1)
> identify(time, fitted(fit2), dimnames(hills)[[1]])
> plot(fitted(fit2), res)
> identify(fitted(fit2), res, dimnames(hills)[[1]])
```

Il secondo grafico evidenzia ancora le osservazioni con residui particolarmente elevati. Potrebbe trattarsi di osservazioni anomale. Per valutare l'effetto di queste osservazioni, possiamo ripetere l'analisi eliminandole, una alla volta.

Esercizio: Ripetere l'adattamento del modello eliminando, nell'ordine, l'osservazione 18 e l'osservazione 11. Che effetto ha sulle stime e sull'adattamento generale del modello l'eliminazione delle due osservazioni?

```
> detach()
```

7.2 Analisi del dataset GASOLINE.DAT

Le variabili del dataset `gasoline.dat` rappresentano rispettivamente:

- y percentuale di benzina ottenuta dal petrolio grezzo;
- x_1 peso del petrolio (?);
- x_2 pressione del petrolio allo stato gassoso;
- x_3 temperatura alla quale il 10% di petrolio passa allo stato gassoso;
- x_4 temperatura di passaggio di tutto il petrolio allo stato gassoso.

Si vuole proporre un modello per la dipendenza di y dalle altre variabili, opportunamente selezionate.

```
> gaso<-read.table("I:\\modelli\\gasoline.dat",
+ col.names=c("y","x1","x2","x3","x4"))
> attach(gaso)
> gaso
      y   x1  x2  x3  x4
 1  6.9 38.4 6.1 220 235
.....
32 45.7 50.8 8.6 190 407
```

Procediamo all'analisi preliminare dei dati:

```
> par(mfrow=c(2,2))
> plot(x1,y)
> plot(x2,y)
> plot(x3,y)
> plot(x4,y)
```

I grafici sono indicativi solo della relazione fra la variabile risposta e e ciascuna delle variabili esplicative, ma non danno informazione globale sulla dipendenza di y da tutte le variabili esplicative simultaneamente. Un comando per vedere simultaneamente i grafici di tutte le possibili coppie di variabili è

```
> par(mfrow=c(1,1))
> pairs(gaso)
```

Soffermiamoci in particolare sui grafici fra coppie di variabili esplicative.

```
> plot(x2,x3)
> plot(x1,x3)
> plot(x1,x2)
```

Si osserva una dipendenza lineare abbastanza marcata fra x_2 e x_3 e fra x_1 e x_3. Meno evidente è la relazione fra x_1 e x_2. Vediamo, quindi sempre a livello esplorativo, le correlazioni tra le variabili.

```
> cor(gaso)
```

Il comando produce la matrice di correlazione tra tutte le variabili. Per quanto riguarda la relazione tra risposta ed esplicative, la correlazione più alta è quella fra x_4 ed y, come già ci si aspettava dai grafici precedenti. Tra le variabili esplicative, le correlazioni più alte sono quelle tra x_2 e x_3, tra x_1 e x_3 e tra x_1 e x_2. La possibile dipendenza fra le variabili esplicative, per esempio fra x_2 e x_3, comporta problemi legati al fatto che nella matrice di regressione le colonne relative a x_2 ed x_3 sono quasi "linearmente dipendenti". In questi casi si parla di multicollinearità ed ha l'effetto di rendere le stime numericamente instabili. Da quanto osservato ci si aspetta che un buon modello non includa coppie di variabili esplicative fortemente correlate (come x_2 e x_3).
Procediamo allora alla costruzione del modello. La prima variabile da includere è x_4 che ha il coefficiente di correlazione più elevato con la variabile risposta.

```
> fit4<-lm(y~x4)
> summary(fit4)

Call:
lm(formula = y ~ x4)

Residuals:
      Min        1Q    Median        3Q       Max
-14.75837  -6.27829   0.05255   5.16243  17.84805

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -16.66206    6.68721  -2.492   0.0185 *
x4            0.10937    0.01972   5.546 4.98e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 7.659 on 30 degrees of freedom
Multiple R-Squared: 0.5063,     Adjusted R-squared: 0.4898
F-statistic: 30.76 on 1 and 30 degrees of freedom,      p-value: 4.983e-006
```

Per la scelta della seconda variabile da introdurre, utilizziamo la funzione `anova` che permette di confrontare due modelli annidati. Confronteremo allora ciascuno dei tre modelli ottenuti aggiungendo a `fit4` una delle prime tre variabili esplicative, con `fit4` stesso. Sceglieremo poi la variabile dalla quale si è ottenuto il modello più significativo.
I tre modelli sono:

```
> fit41<-update(fit4,.~.+x1)
> fit42<-update(fit4,.~.+x2)
> fit43<-update(fit4,.~.+x3)
```

Dal confronto con il modello con la sola x_4 si ottiene:

```
> anova(fit4,fit41)
Analysis of Variance Table

Model 1: y ~ x4
Model 2: y ~ x4 + x1
  Res.Df Res.Sum Sq Df  Sum Sq F value  Pr(>F)
1     30    1759.69
2     29     861.95  1  897.75  30.204 6.4e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
> anova(fit4,fit42)
Analysis of Variance Table

Model 1: y ~ x4
Model 2: y ~ x4 + x2
  Res.Df Res.Sum Sq Df  Sum Sq F value    Pr(>F)
1     30    1759.69
2     29     369.87  1 1389.83  108.97 2.468e-11 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
> anova(fit4,fit43)
Analysis of Variance Table

Model 1: y ~ x4
Model 2: y ~ x4 + x3
  Res.Df Res.Sum Sq Df  Sum Sq F value    Pr(>F)
1     30    1759.69
2     29     170.61  1 1589.08  270.11 3.331e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Si sceglie allora il modello per cui si è ottenuta la somma dei quadrati dei residui (`Res.Sum Sq`) più bassa, o equivalentemente la somma dei quadrati di regressione (`Sum Sq`) e il valore F (`F value`) più alti. Il modello migliore è `fit43`, cioé quello che include le variabili x_4 e x_3.

```
> summary(fit43)

Call:
lm(formula = y ~ x4 + x3)

Residuals:
    Min      1Q  Median      3Q     Max
-3.9593 -1.9063 -0.3711  1.6242  4.3802

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 18.467633   3.009009   6.137 1.09e-06 ***
x4           0.155813   0.006855  22.731  < 2e-16 ***
x3          -0.209329   0.012737 -16.435 2.22e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.426 on 29 degrees of freedom
Multiple R-Squared: 0.9521,     Adjusted R-squared: 0.9488
F-statistic: 288.4 on 2 and 29 degrees of freedom,      p-value:     0
```

Ripetiamo ora lo stesso procedimento per scegliere fra x_1 e x_2.

```
> fit431<-update(fit43,.~.+x1)
> fit432<-update(fit43,.~.+x2)

> anova(fit43,fit431)
Analysis of Variance Table

Model 1: y ~ x4 + x3
Model 2: y ~ x4 + x3 + x1
  Res.Df Res.Sum Sq Df Sum Sq F value  Pr(>F)
1     29     170.61
2     28     146.00  1  24.61  4.7198 0.03844 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


> anova(fit43,fit432)
Analysis of Variance Table

Model 1: y ~ x4 + x3
Model 2: y ~ x4 + x3 + x2
  Res.Df Res.Sum Sq Df  Sum Sq F value Pr(>F)
1     29    170.612
2     28    160.620  1   9.992  1.7419 0.1976
```

I risultati portano a scegliere x_1. In particolare, dal p-value del secondo confronto (0.1976) si deduce che il modello non migliora se si aggiunge la variabile x_2. Questo ce lo aspettavamo, visto che il modello include già x_3 e x_3 e x_2 sono fortemente correlate. Il p-value relativo al primo confronto (0.03844) è comunque indice di un test di livello 0.05 significativo. Vediamo il `summary` corrispondente al modello con x_4, x_3, x_1:

```
> summary(fit431)


Call:
lm(formula = y ~ x4 + x3 + x1)

Residuals:
    Min      1Q  Median      3Q     Max
-3.5303 -1.3606 -0.2681  1.3911  4.7658

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  4.032034   7.223341   0.558   0.5811
x4           0.156527   0.006462  24.224  < 2e-16 ***
x3          -0.186571   0.015922 -11.718 2.61e-12 ***
x1           0.221727   0.102061   2.173   0.0384 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.283 on 28 degrees of freedom
Multiple R-Squared: 0.959,      Adjusted R-squared: 0.9546
F-statistic: 218.5 on 3 and 28 degrees of freedom,      p-value:     0
```

Se si sceglie dunque di introdurre anche x_1, si può eliminare l'intercetta che ora non è più significativa (p-value=0.5811). Il modello risultante è:

```
> fit431bis<-update(fit431,.~.-1)
> summary(fit431bis)

Call:
lm(formula = y ~ x4 + x3 + x1 - 1)

Residuals:
    Min      1Q  Median      3Q     Max
-3.6075 -1.3229 -0.3831  1.7549  4.9115

Coefficients:
    Estimate Std. Error t value Pr(>|t|)
x4  0.157168   0.006283  25.017  < 2e-16 ***
```

```
x3 -0.179328   0.009116 -19.672  < 2e-16 ***
x1  0.274133   0.039548   6.932 1.28e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.256 on 29 degrees of freedom
Multiple R-Squared: 0.9907,     Adjusted R-squared: 0.9898
F-statistic:  1034 on 3 and 29 degrees of freedom,      p-value:     0
```

Questo potrebbe essere un modello adeguato. Ma si può anche decidere di non considerare la variabile x_1 significativa e proporre invece il modello `fit43` con x_4, x_3 e l'intercetta.
Un procedimento alternativo per la selezione delle variabili da includere nel modello è il seguente. Si comincia con il modello lineare che include tutte le variabili esplicative.

```
> fit<-lm(y~x1+x2+x3+x4)
> summary(fit)


Call:
lm(formula = y ~ x1 + x2 + x3 + x4)

Residuals:
    Min      1Q  Median      3Q     Max
-3.5804 -1.5223 -0.1098  1.4237  4.6214

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -6.820774  10.123152  -0.674   0.5062
x1           0.227246   0.099937   2.274   0.0311 *
x2           0.553726   0.369752   1.498   0.1458
x3          -0.149536   0.029229  -5.116 2.23e-05 ***
x4           0.154650   0.006446  23.992  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.234 on 27 degrees of freedom
Multiple R-Squared: 0.9622,     Adjusted R-squared: 0.9566
F-statistic: 171.7 on 4 and 27 degrees of freedom,      p-value:     0
```

Dai livelli di significatività osservati relativi ai test sui singoli parametri, si nota che ci sono coefficienti non significativi. Le variabili non significative si eliminano una alla volta, cominciando da quella in corrispondenza della quale si è osservato un p-value molto alto, in questo caso l'intercetta.

```
> fit1<-update(fit,.~.-1)
> summary(fit1)

Call:
lm(formula = y ~ x1 + x2 + x3 + x4 - 1)

Residuals:
    Min      1Q  Median      3Q     Max
-3.6693 -1.2920 -0.1271  1.2348  4.5478

Coefficients:
    Estimate Std. Error t value Pr(>|t|)
x1  0.182249   0.073618   2.476   0.0196 *
x2  0.375377   0.255639   1.468   0.1531
x3 -0.167438   0.012061 -13.882 4.44e-14 ***
x4  0.154725   0.006382  24.245  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.213 on 28 degrees of freedom
Multiple R-Squared: 0.9914,     Adjusted R-squared: 0.9902
F-statistic: 806.6 on 4 and 28 degrees of freedom,      p-value:     0
```

Il coefficiente di x_2 è ancora non significativo. Escludendo anche x_2, otteniamo esattamente il modello `fit431bis`.
Si può ora passare all'analisi dei residui per il modello `fit431bis`.

```
> res<-rstandard(fit431bis)
> par(mfrow=c(2,1))
> hist(res)
> boxplot(res)
> par(mfrow=c(1,1))
> qqnorm(res)
> qqline(res)
```

I residui non presentano andamenti sistematici e, fatta eccezione per una leggera asimmetria, possono essere considerati normali. Il modello si adatta bene ai dati. Per vedere se rimane qualche tipo di dipendenza dalle variabili esplicative:

```
> plot(x1,res)
> plot(x3,res)
> plot(x4,res)
```

Per assicurarci di aver eliminato effettivamente una variabile non importante:

```
> plot(x2,res)
```

Esercizio: analizzare i residui del modello `fit43`.

```
> detach()
```