Materiale didattico per i laboratori di Modelli Statistici I[1]

M. Chiogna, A. Salvan e N. Sartori

Anno Accademico 2006-2007

[1]Documento preparato con LaTeX, Sweave e R 2.0.0 su i386-pc-mingw32 in data 5 dicembre 2006.

Indice

Capitolo 1

Richiami di R

1.1 Funzioni e comandi elementari

R è un ambiente di *software* integrato per la manipolazione di dati, il calcolo e la rappresentazione grafica. Per iniziare una sessione, è necessario effettuare un doppio click di *mouse* sulla icona di R. Si aprirà in questo modo la finestra di comando e verrà proposto il *prompt* di comando:

```
>
```

Le entità che R crea durante una sessione di lavoro sono chiamate *oggetti*. Questi possono essere numeri, stringhe, vettori, matrici, funzioni, o strutture più generali costruite da questi elementi. Tali oggetti sono salvati per nome e immagazzinati in un'area dedicata detta *workspace* o spazio di lavoro. In ogni momento, è possibile controllare gli oggetti disponibili nello spazio di lavoro mediante il comando

```
> ls()
```

Per eliminare un oggetto dallo spazio di lavoro, si usa la funzione `rm()`. La funzione prevede come argomento il nome dell'oggetto che si vuole eliminare. Supponendo che sia presente un oggetto di nome `thing`, è possibile eliminarlo con il comando

```
> rm(thing)
```

A questo punto, l'oggetto di nome `thing` non sarà più presente nel *workspace*

```
> thing
```

```
Error:  Object thing not found
```

Se si vogliono eliminare più oggetti, bisogna elencarli separati da virgole.

```
> rm(thing1, thing2)
```

Quando si inizia una nuova sessione di lavoro, è opportuno rimuovere gli oggetti esistenti nell'area di lavoro. Un comando utile a tale scopo è

```
> rm(list = ls())
```

o, in alternativa, `rm(list=objects())`.
Per ottenere informazioni su una funzione di R si può utilizzare la funzione `help()`, specificando come argomento il nome della funzione di interesse.
Per terminare una sessione aperta, si usa la funzione `q()`. Alla chiusura della sessione, è possibile salvare gli oggetti R disponibili nello spazio di lavoro, se si intende riutilizzali nelle sessioni di lavoro future. Per salvare tali oggetti, è necessario rispondere affermativamente alla domanda proposta in chiusura da R.
I comandi elementari consistono in espressioni o assegnazioni. Se il comando è una espressione, R fornisce il risultato della valutazione, come è illustrato negli esempi che seguono.

```
> 12 > 10

[1] TRUE

> 1 + 2 + 3

[1] 6

> 2 + 3 * 4

[1] 14

> 3/2 + 1

[1] 2.5

> 2 + (3 * 4)

[1] 14

> (2 + 3) * 4

[1] 20

> 4 * 3^3

[1] 108
```

Tutte le funzioni matematiche generalmente presenti su una calcolatrice tascabile sono disponibili in R sotto forma di funzioni di base, richiamate nella tabella che segue.

Funzione R	Funzione matematica
`sqrt`	radice quadrata
`abs`	valore assoluto
`sin,cos,tan`	funzioni trigonometriche
`asin,acos,atan`	funzioni trigonometriche inverse
`exp,log`	exponenziale e logaritmo naturale

L'uso di tali funzioni, che può essere anche annidato, è elementare.

```
> sqrt(2)

[1] 1.414214

> sin(3.14159)

[1] 2.65359e-06

> sin(pi)

[1] 1.224606e-16

> sqrt(sin(45 * pi/180))

[1] 0.8408964
```

Una assegnazione valuta una espressione salvandone l'esito in un oggetto dotato di nome. L'assegnazione viene effettuata mediante il simbolo `<-` , oppure il simbolo `=`. Si può anche assegnare da sinistra verso destra con il simbolo `->`. Il risultato di una assegnazione non è automaticamente mostrato. L'utente può visualizzarlo richiamando il nome dell'oggetto. Gli oggetti creati mediante un'assegnazione possono essere riutilizzati in espressioni e assegnazioni successive, come mostrato negli esempi che seguono.

```
> x <- sqrt(2)
> x

[1] 1.414214

> x^3

[1] 2.828427

> y <- x^3
> x <- 10
> x > 10

[1] FALSE

> x <= 10

[1] TRUE

> tf <- x > 10
> tf

[1] FALSE
```

1.2 Vettori

1.2.1 Creazione

Per creare un vettore, si usa la funzione `c()`:

```
> x <- c(2, 3, 5, 7, 11)
> x

[1]  2  3  5  7 11
```

Se il vettore contiene tanti elementi, può essere più conveniente usare la funzione `scan()`, che consente di introdurli uno ad uno.

```
> x <- scan()

1: 1
2: 6
3: 3
4: 4
5:
> x
[1] 1 6 3 4
>

> x <- scan()

1: 23 34 32
4: 33 88 44
7:
```

Per creare una successione di numeri da `a` a `b`, si può usare il comando `a:b`.

```
> xx <- 1:10
> xx

 [1]  1  2  3  4  5  6  7  8  9 10

> xx <- 100:1
> xx

  [1] 100  99  98  97  96  95  94  93  92  91  90  89  88
 [14]  87  86  85  84  83  82  81  80  79  78  77  76  75
 [27]  74  73  72  71  70  69  68  67  66  65  64  63  62
 [40]  61  60  59  58  57  56  55  54  53  52  51  50  49
 [53]  48  47  46  45  44  43  42  41  40  39  38  37  36
 [66]  35  34  33  32  31  30  29  28  27  26  25  24  23
 [79]  22  21  20  19  18  17  16  15  14  13  12  11  10
 [92]   9   8   7   6   5   4   3   2   1
```

Lo stesso risultato può essere ottenuto tramite la funzione `seq()`.

```
> xx <- seq(from = 100, to = 1)
> xx

  [1] 100  99  98  97  96  95  94  93  92  91  90  89  88
 [14]  87  86  85  84  83  82  81  80  79  78  77  76  75
 [27]  74  73  72  71  70  69  68  67  66  65  64  63  62
 [40]  61  60  59  58  57  56  55  54  53  52  51  50  49
 [53]  48  47  46  45  44  43  42  41  40  39  38  37  36
 [66]  35  34  33  32  31  30  29  28  27  26  25  24  23
 [79]  22  21  20  19  18  17  16  15  14  13  12  11  10
 [92]   9   8   7   6   5   4   3   2   1
```

Possono anche essere creati vettori che contengono elementi ripetuti.

```
> rep(2, times = 3)

[1] 2 2 2

> rep(2, 3)

[1] 2 2 2

> a = c(rep(2, 3), 4, 5, rep(1, 5))
> a

 [1] 2 2 2 4 5 1 1 1 1 1
```

Ai vettori può essere applicata l'aritmetica di base e le operazioni logiche che si applicano agli scalari.

```
> x <- 1:10
> x * 2

 [1]  2  4  6  8 10 12 14 16 18 20

> x * x

 [1]   1   4   9  16  25  36  49  64  81 100

> x > 5

 [1] FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE  TRUE  TRUE
[10]  TRUE
```

Alcune funzioni utili per la manipolazione di vettori sono elencate di seguito.

```
> x <- 3:26
> length(x)
```

```
[1] 24

> max(x)

[1] 26

> min(x)

[1] 3

> sum(x)

[1] 348

> prod(x)

[1] 2.016457e+26

> mean(x)

[1] 14.5

> var(x)

[1] 50

> range(x)

[1]  3 26
```

1.2.2 Estrazione di elementi

Gli elementi di un vettore possono essere estratti usando le parentesi quadre [ ] ed indicando tra parentesi la posizione dell'elemento che si vuole estrarre.

```
> xx[7]
[1] 94
```

Si possono estrarre anche più elementi contemporaneamente.

```
> xx[c(2, 3, 5, 7, 11)]

[1] 99 98 96 94 90

> xx[85:91]

[1] 16 15 14 13 12 11 10

> xx[91:85]

[1] 10 11 12 13 14 15 16
```

```
> xx[c(1:5, 8:10)]

[1] 100  99  98  97  96  93  92  91

> xx[c(1, 1, 1, 1, 2, 2, 2, 2)]

[1] 100 100 100 100  99  99  99  99
```

Ovviamente, i sottoinsiemi di elementi estratti possono essere salvati in nuovi vettori.

```
> yy <- xx[c(1, 2, 4, 8, 16, 32, 64)]
> yy

[1] 100  99  97  93  85  69  37
```

Se gli indicatori entro le parentesi quadre sono preceduti dal segno negativo, gli elementi corrispondenti vengono eliminati dal vettore.

```
> x <- c(1, 2, 4, 8, 16, 32)
> x

[1]  1  2  4  8 16 32

> x[-4]

[1]  1  2  4 16 32
```

1.3 Matrici

La matrici vengono create mediante la funzione `matrix()`. Nella sua forma più semplice, l'uso della funzione prevede di specificare un vettore contenente gli elementi della matrice e il numero di righe o di colonne della matrice.

```
> x <- matrix(c(2, 3, 5, 7, 11, 13), nrow = 3)
> x

     [,1] [,2]
[1,]    2    7
[2,]    3   11
[3,]    5   13

> x <- matrix(c(2, 3, 5, 7, 11, 13), ncol = 2)
> x

     [,1] [,2]
[1,]    2    7
[2,]    3   11
[3,]    5   13
```

Ovviamente, la matrice può essere acquisita da un *file* esterno. Si supponga, per esempio, che il *file* `matdata` abbia il seguente contenuto.

```
1,24,32,36,33
2,16,44,34,33
3,20,31,43,32
4,23,35,37,35
5,27,40,40,31
6,19,43,32,37
```

Il contenuto del *file* può essere acquisito ed assegnato ad una matrice 6×5 con i comandi:

```
> x2 <- scan("matdata", sep = ",")

> mx <- matrix(x2, ncol = 5, byrow = TRUE)
> mx

     [,1] [,2] [,3] [,4] [,5]
[1,]    1   24   32   36   33
[2,]    2   16   44   34   33
[3,]    3   20   31   43   32
[4,]    4   23   35   37   35
[5,]    5   27   40   40   31
[6,]    6   19   43   32   37
```

La funzione `dim()` restituisce la dimensione (numero di righe e numero di colonne) della matrice indicata come argomento.

```
> dim(mx)

[1] 6 5
```

Come per i vettori, gli elementi di una matrice possono essere estratti mediante l'uso delle parentesi `[]`. Per estrarre da una matrice un elemento, bisogna specificarne la posizione di riga e di colonna.

```
> x[2, 1]

[1] 3

> x[2, 2]

[1] 11
```

Per estrarre una intera riga o colonna, è sufficiente specificarne la posizione.

```
> x[, 1]

[1] 2 3 5
```

```
> x[3, ]

[1]  5 13
```

Possono essere estratti sottoinsiemi di righe e/o colonne.

```
> x <- matrix(1:16, ncol = 4)
> x

     [,1] [,2] [,3] [,4]
[1,]    1    5    9   13
[2,]    2    6   10   14
[3,]    3    7   11   15
[4,]    4    8   12   16

> x[c(1, 4), c(3, 4)]

     [,1] [,2]
[1,]    9   13
[2,]   12   16
```

1.4 *Data frame*

Un *data frame* è un oggetto simile ad una matrice usato per rappresentare una matrice di dati. In un *data frame*, ogni riga rappresenta una unità statistica ed ogni colonna rappresenta una variabile. Le colonne possono contenere variabili quantitative o qualitative.
Per leggere una matrice di dati da *file* esterno, si usa la funzione `read.table()`, che automaticamente controlla se le variabili sono quantitative o qualitative e se le righe/colonne hanno etichette. Si supponga che nella *directory* di lavoro si trovi il *file*, di nome `cherry.dat`, così costituito:

```
8.3     70      10.3
8.6     65      10.3
8.8     63      10.2
10.5    72      16.4
10.7    81      18.8
10.8    83      19.7
...
...
```

Si noti, in particolare, che il *file* non contiene alcuna intestazione, ovvero non è specificato alcun nome per le righe e le colonne del *file*. Si può acquisire il contenuto del *file* e assegnarlo ad un *data frame* mediante il comando:

```
> Ciliegi <- read.table("cherry.dat")
```

Se il *file* `cherry.dat` si trova in una *directory* diversa da quella di lavoro, ad esempio in `I:/modelli I`, bisogna dare il percorso completo, ad esempio

```
> Ciliegi <- read.table("I:/modelli I/cherry.dat")
```

Un'altra possibilità, particolarmente utile quando non si ricorda l'esatto percorso del *file*, è utilizzare la funzione `file.choose()` all'interno di `read.table()`.

```
> Ciliegi <- read.table(file.choose())
```

In questo modo, è possibile selezionare il file desiderato in modo interattivo.

```
> Ciliegi <- read.table("cherry.dat")
> Ciliegi
```

Non essendovi etichette, `R` automaticamente assegna i nomi alle variabili (cioè alle colonne) presenti nel *data frame*. Questi sono `V1`, `V2` e `V3`.

```
> names(Ciliegi)

[1] "V1" "V2" "V3"
```

Tali nomi possono essere modificati mediante la funzione `names()`.

```
> names(Ciliegi) <- c("diametro", "altezza", "volume")
```

Alternativamente, i nomi delle variabili possono essere specificati direttamente in fase di lettura, utilizzando l'argomento `col.names` della funzione `read.table()`, come mostrato nel seguito.

```
> Ciliegi <- read.table("I:/modelli I/cherry.dat",
+     col.names = c("diametro", "altezza", "volume"))
```

Statistiche riassuntive sulle variabili contenute in un *data frame* possono essere ottenute mediante la funzione `summary()`.

```
> summary(Ciliegi)

    diametro        altezza       volume
 Min.   : 8.30   Min.   :63   Min.   :10.20
 1st Qu.:11.05   1st Qu.:72   1st Qu.:19.40
 Median :12.90   Median :76   Median :24.20
 Mean   :13.25   Mean   :76   Mean   :30.17
 3rd Qu.:15.25   3rd Qu.:80   3rd Qu.:37.30
 Max.   :20.60   Max.   :87   Max.   :77.00
```

Il *data frame* è anche una matrice, quindi un oggetto dotato di due dimensioni.

```
> dim(Ciliegi)

[1] 31  3
```

Da un *data frame*, dunque, possono essere estratti singoli elementi, variabili (colonne), unità statistiche (righe) e sottoinsiemi di elementi utilizzando la stessa sintassi introdotta per l'estrazione di elementi dalle matrici.

```
> Ciliegi[, 3]

 [1] 10.3 10.3 10.2 16.4 18.8 19.7 15.6 18.2 22.6 19.9 24.2
[12] 21.0 21.4 21.3 19.1 22.2 33.8 27.4 25.7 24.9 34.5 31.7
[23] 36.3 38.3 42.6 55.4 55.7 58.3 51.5 51.0 77.0
```

Tuttavia, un *data frame* permette di estrarre la variabili anche utilizzandone il nome, mediante l'operatore dollaro.

```
> Ciliegi$volume

 [1] 10.3 10.3 10.2 16.4 18.8 19.7 15.6 18.2 22.6 19.9 24.2
[12] 21.0 21.4 21.3 19.1 22.2 33.8 27.4 25.7 24.9 34.5 31.7
[23] 36.3 38.3 42.6 55.4 55.7 58.3 51.5 51.0 77.0
```

Quando, durante una sessione di lavoro, si vogliono effettuare operazioni ripetute sullo stesso *data frame*, per esempio il *data frame* `Ciliegi`, è possibile comunicare ad `R` che i comandi successivi sono da applicare al *data frame* selezionato. Ciò può essere fatto utilizzando la funzione `attach()`.

```
> attach(Ciliegi)
```

Dopo aver effettuato il comando, tutti le operazioni vengono eseguite sul *data frame* `Ciliegi`. Per esempio, per estrarre dal *data frame* la variabile `volume`, è sufficiente digitare il nome della variabile.

```
> volume

 [1] 10.3 10.3 10.2 16.4 18.8 19.7 15.6 18.2 22.6 19.9 24.2
[12] 21.0 21.4 21.3 19.1 22.2 33.8 27.4 25.7 24.9 34.5 31.7
[23] 36.3 38.3 42.6 55.4 55.7 58.3 51.5 51.0 77.0
```

Per le variabili estratte, possono essere costruite rappresentazioni grafiche. I seguenti comandi producono i grafici nella Figura 1.1

```
> hist(diametro)

> boxplot(diametro)
```

L'estrazione di elementi da un *data frame* per cui sia stato eseguito il comando `attach`, segue le stesse regole dell'estrazione di elementi da matrici. Ad esempio, il comando seguente consente di ottenere il *data frame* relativo agli alberi con altezza superiore a 80 piedi.

```
> Ciliegi[altezza > 80, ]
```

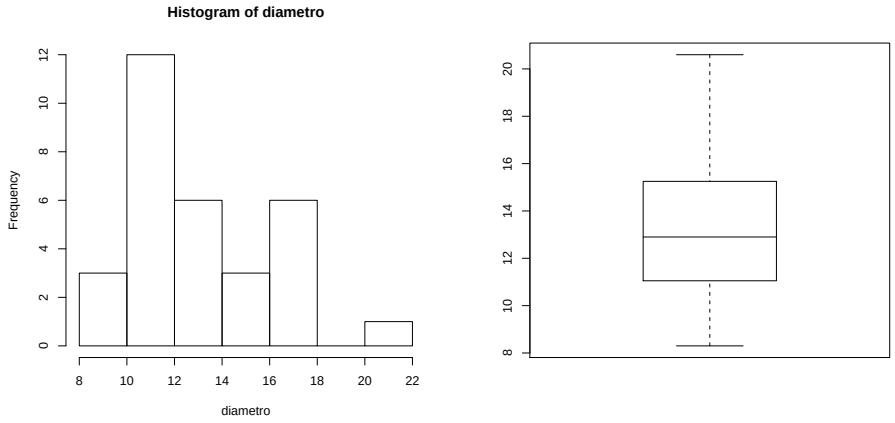


Figura 1.1: Istogramma (sinistra) e boxplot (destra) di `diametro`

```
   diametro altezza volume
5      10.7      81   18.8
6      10.8      83   19.7
17     12.9      85   33.8
18     13.3      86   27.4
26     17.3      81   55.4
27     17.5      82   55.7
31     20.6      87   77.0
```

Gli effetti della funzione `attach()` si annullano mediante la funzione `detach()`.

```
> detach(Ciliegi)
```

Capitolo 2

Modello di regressione lineare semplice

2.1 Analisi dei dati CHERRY.DAT

Il *file* `cherry.dat` contiene i dati relativi a 31 alberi di ciliegio abbattuti. In particolare, la misura del volume di legno ricavato dall'albero (in piedi cubi), il diametro (in pollici) del tronco misurato a poco più di un metro dal suolo e l'altezza (in piedi) dell'albero. Si noti che un pollice corrisponde a 0.0833 piedi e un piede corrisponde a 30.48 centimetri.
Occorre acquisire i dati in `R`.

```
> Ciliegi <- read.table(file.choose(), col.names = c("diametro",
+     "altezza", "volume"))
```

```
> attach(Ciliegi)
```

L'oggetto `Ciliegi` si trova ora nello spazio di lavoro e le variabili `volume`, `diametro` e `altezza` sono direttamente accessibili.
Si desidera in primo luogo studiare la relazione tra il volume di legno e il diametro. La dipendenza dal volume da entrambe le variabili esplicative (diametro e altezza), sarà studiata nel Paragrafo 7.1. Siano (x_i, y_i), $i = 1, \ldots, 31$, i dati relativi alle due variabili diametro e volume. Si ottiene il diagramma di dispersione delle coppie (x_i, y_i), $i = 1, \ldots, 31$, mostrato in Figura 2.1, con il comando

```
> plot(diametro, volume)
```

Il numero di osservazioni è

```
> n <- nrow(Ciliegi)
```

Per spiegare la relazione tra volume e diametro, si assume che $y_1, \ldots, y_{31}$ siano realizzazioni di variabili casuali Y_i dove

$$Y_i = \beta_1 + \beta_2 x_i + \varepsilon_i \,, \tag{2.1}$$

con ε_i variabili casuali $N(0, \sigma^2)$ indipendenti, $i = 1, \ldots, 31$.
La stima di massima verosimiglianza (e con il metodo dei minimi quadrati), $\hat{\beta}_2$, di β_2 è

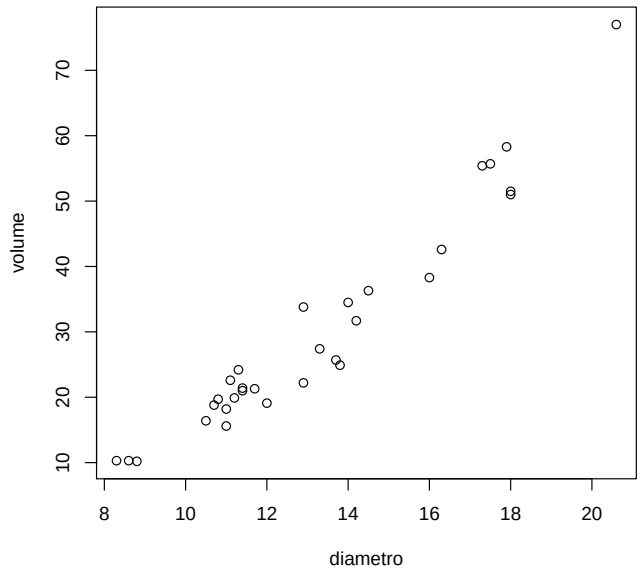


Figura 2.1: Diagramma di dispersione di `diametro` e `volume`.

```
> beta2.hat <- (sum(diametro * volume)/n - mean(volume) *
+     mean(diametro))/(mean(diametro^2) - mean(diametro)^2)
> beta2.hat

[1] 5.065856
```

Lo stesso risultato si ottiene con

```
> beta2.hat <- cov(diametro, volume)/var(diametro)
> beta2.hat

[1] 5.065856
```

La stima, $\hat{\beta}_1$, di β_1 è pari a

```
> beta1.hat <- mean(volume) - beta2.hat * mean(diametro)
> beta1.hat

[1] -36.94346
```

Si può aggiungere la retta stimata al diagramma di dispersione con il comando

```
> abline(beta1.hat, beta2.hat, lty = "dashed")
```

ottenendo la Figura 2.2. Il comando `abline(a,b)` traccia una retta nel grafico corrente con intercetta `a` e coefficiente angolare `b`. L'argomento `lty` specifica un'opzione grafica generale (valida ad esempio anche per il comando `plot`) che definisce il tipo

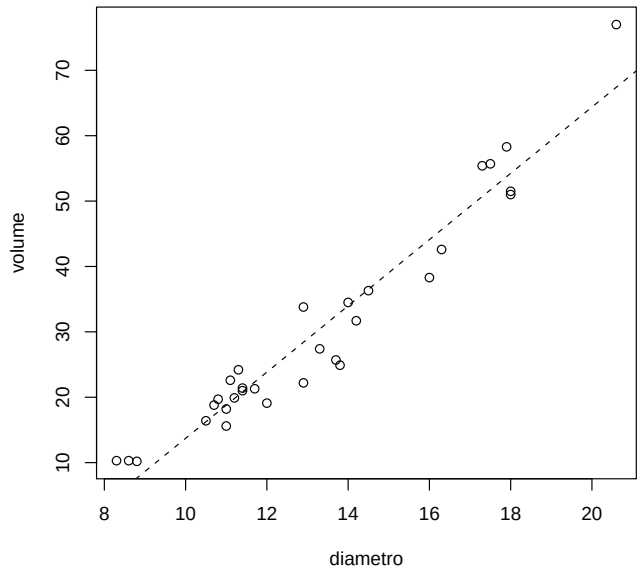


Figura 2.2: Diagramma di dispersione di `diametro` e `volume` e retta di regressione stimata.

di linea. Assume valori: "blank", "solid" (default), "dashed", "dotted", "dotdash", "longdash" o "twodash", oppure i numeri da 0 a 7 corrispondenti.

I valori predetti, o stimati, dal modello sono $\hat{y}_i = \hat{\beta}_1 + \hat{\beta}_2 x_i$ e si ottengono con

```
> valori.predetti <- beta1.hat + beta2.hat * diametro
> valori.predetti

 [1]  5.103149  6.622906  7.636077 16.248033 17.261205
 [6] 17.767790 18.780962 18.780962 19.287547 19.794133
[11] 20.300718 20.807304 20.807304 22.327061 23.846818
[16] 28.406089 28.406089 30.432431 32.458774 32.965360
[21] 33.978531 34.991702 36.511459 44.110244 45.630001
[26] 50.695857 51.709028 53.735371 54.241956 54.241956
[31] 67.413183
```

Ovviamente, i valori predetti stanno sulla retta di regressione. Si possono aggiungere questi punti al grafico precedente con il comando `points()`.

```
> points(diametro, valori.predetti, pch = "X")

NULL
```

ottenendo il grafico mostrato in Figura 2.3
L'argomento `pch` permette di scegliere il tipo di carattere da utilizzare nel grafico per identificare un punto (in questo caso si è scelto X).
I residui sono dati dalla differenza tra i valori osservati e quelli stimati, $e_i = y_i - \hat{y}_i$,

```
NULL
```

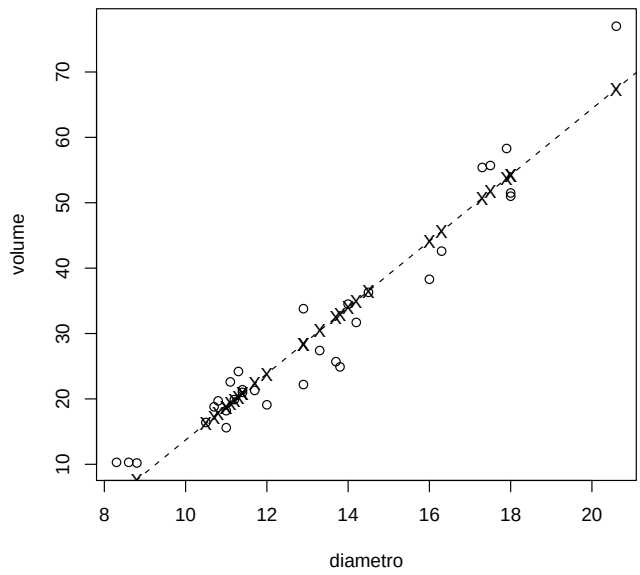


Figura 2.3: Diagramma di dispersione di `diametro` e `volume`, retta di regressione stimata e valori predetti dal modello.

```
> residui <- volume - valori.predetti
> residui

 [1]  5.1968508  3.6770939  2.5639226  0.1519667  1.5387954
 [6]  1.9322098 -3.1809615 -0.5809615  3.3124528  0.1058672
[11]  3.8992815  0.1926959  0.5926959 -1.0270610 -4.7468179
[16] -6.2060887  5.3939113 -3.0324313 -6.7587739 -8.0653595
[21]  0.5214692 -3.2917021 -0.2114590 -5.8102436 -3.0300006
[26]  4.7041430  3.9909717  4.5646292 -2.7419565 -3.2419565
[31]  9.5868168
```

Il coefficiente di determinazione R^2 è dato da

```
> R2 <- 1 - var(residui)/var(volume)
> R2

[1] 0.9353199
```

La stima di massima verosimiglianza di σ^2 è data da

```
> sigma2.hat <- sum(residui^2)/n
> sigma2.hat

[1] 16.91299
```

Si può calcolare la stima non distorta di σ^2

```
> s2 <- sum(residui^2)/(n - 2)
> s2
```

```
[1] 18.07940
```

che è equivalente a

```
> sigma2.hat * n/(n - 2)
```

```
[1] 18.07940
```

Utilizzando `s2` si possono calcolare le stime non distorte $\hat{V}(\hat{\beta}_1)$ e $\hat{V}(\hat{\beta}_2)$ delle varianze di $\hat{\beta}_1$ e $\hat{\beta}_2$

$$\begin{aligned} \hat{V}(\hat{\beta}_1) &= s^2 \left(\frac{1}{n} + \frac{\bar{x}^2}{\sum_{i=1}^n (x_i - \bar{x})^2} \right) \\ \hat{V}(\hat{\beta}_2) &= \frac{s^2}{\sum_{i=1}^n (x_i - \bar{x})^2} \end{aligned}$$

```
> var.beta1.hat <- s2 * (1/n + mean(diametro)^2/sum((diametro -
+     mean(diametro))^2))
> var.beta1.hat
```

```
[1] 11.3242
```

```
> var.beta2.hat <- s2/sum((diametro - mean(diametro))^2)
> var.beta2.hat
```

```
[1] 0.06119536
```

L'estremo inferiore e superiore di un intervallo di confidenza con livello esatto 0.95 per β_1 si calcolano come

```
> beta1.lower <- beta1.hat - qt(0.975, n - 2) *
+     sqrt(var.beta1.hat)
> beta1.upper <- beta1.hat + qt(0.975, n - 2) *
+     sqrt(var.beta1.hat)
> beta1.lower
```

```
[1] -43.82595
```

```
> beta1.upper
```

```
[1] -30.06096
```

dove il comando `qt(p,df)` restituisce il quantile relativo alla probabilità `p` della distribuzione t di Student con `df` gradi di libertà. In modo analogo, si può ottenere un intervallo di confidenza per β_2.

```
> beta2.lower <- beta2.hat - qt(0.975, n - 2) *
+     sqrt(var.beta2.hat)
> beta2.upper <- beta2.hat + qt(0.975, n - 2) *
+     sqrt(var.beta2.hat)
> beta2.lower

[1] 4.559914

> beta2.upper

[1] 5.571799
```

Si possono anche ottenere estremo inferiore e superiore dell'intervallo con una sola istruzione.

```
> beta2.IC <- beta2.hat + c(-1, 1) * qt(0.975, n -
+     2) * sqrt(var.beta2.hat)
> beta2.IC

[1] 4.559914 5.571799
```

Si desidera ora verificare l'ipotesi di nullità di β_2, contro l'alternativa bilaterale. Il valore osservato della statistica test $t_2 = \hat{\beta}_2 / \sqrt{\hat{V}(\hat{\beta}_2)}$ è

```
> t2 <- beta2.hat/sqrt(var.beta2.hat)
> t2

[1] 20.47829
```

Sotto H_0, `T2` è realizzazione di una variabile casuale t_2 con distribuzione t di Student con $n-2$ gradi di libertà, t_{n-2}. Il livello di significatività osservato,

$$\alpha^{oss} = 2 \min\{Pr(T_2 \geq t_2), Pr(T_2 \leq t_2)\} = 2Pr(T_2 \geq |t_2|),$$

è quindi

```
> 2 * min(pt(t2, n - 2), pt(t2, n - 2, lower.tail = FALSE))

[1] 8.644334e-19
```

o, equivalentemente,

```
> 2 * pt(abs(t2), n - 2, lower.tail = FALSE)

[1] 8.644334e-19
```

È stato utilizzato il comando `pt(q,df)`, che calcola la funzione di ripartizione della distribuzione t di Student con `df` gradi di libertà nel punto `q`. Con l'argomento `lower.tail=FALSE` si ottiene la probabilità sulla coda destra, anziché su quella sinistra.
Il valore ottenuto del livello di significatività osservato è praticamente nullo, indicando che β_2 è significativamente diverso da zero. Un test con livello fissato 0.05, confronta $|t_2^{oss}|$ con il quantile di livello 0.975 di una t_{n-2}, $t_{n-2;0.975}$, che in questo caso è pari a 2.0452, ottenuto con `qt(0.975,n-2)`. Essendo $|t_2^{oss}|$ maggiore di 2.0452, l'ipotesi nulla viene rifiutata, come si può dedurre confrontando il livello di significatività osservato con 0.05.

Esercizio. Si verifichi l'ipotesi di nullità di β_1. Si verifichi inoltre l'ipotesi che β_2 sia uguale a 5. ◇

Le analisi condotte fino ad ora si basano sul modello (2.1). In realtà, il diagramma di dispersione in Figura 2.1 potrebbe essere compatibile con un andamento curvilineo. In effetti, è ragionevole ipotizzare che il volume sia proporzionale al quadrato del diametro:

$$\text{volume} \doteq k \cdot \text{diametro}^2 \,,$$

Si possono allora considerare le trasformazioni logaritmiche delle variabili `volume` e `diametro`, ossia utilizzare come modello di riferimento

$$Z_i = \log Y_i = \beta_1^l + \beta_2^l \log x_i + \varepsilon_i \,, i = 1, \ldots, 31 \,.$$

Le stime dei coefficienti β_1^l e β_2^l sono

```
> beta2.l.hat <- cov(log(volume), log(diametro))/var(log(diametro))
> beta1.l.hat <- mean(log(volume)) - beta2.l.hat *
+     mean(log(diametro))
```

I valori predetti $\hat{z}_i$ si ottengono con

```
> valori.predetti.l <- beta1.l.hat + beta2.l.hat *
+     log(diametro)
```

Si possono ottenere i valori predetti nella scala originale delle variabili con

```
> valori.predetti1 <- exp(valori.predetti.l)
```

Si possono confrontare graficamente i risultati con quelli precedenti (cfr Figura 2.4).

```
> plot(diametro, volume)
> abline(beta1.hat, beta2.hat, lty = "dashed")
> lines(diametro, valori.predetti1)
```

La linea continua sembra adattarsi meglio alle osservazioni negli estremi. La funzione `lines()` consente di sovrapporre linee ad un grafico esistente (si veda `help(lines)` per ulteriori dettagli).
Se si definiscono i residui nella scala originaria delle variabili, come

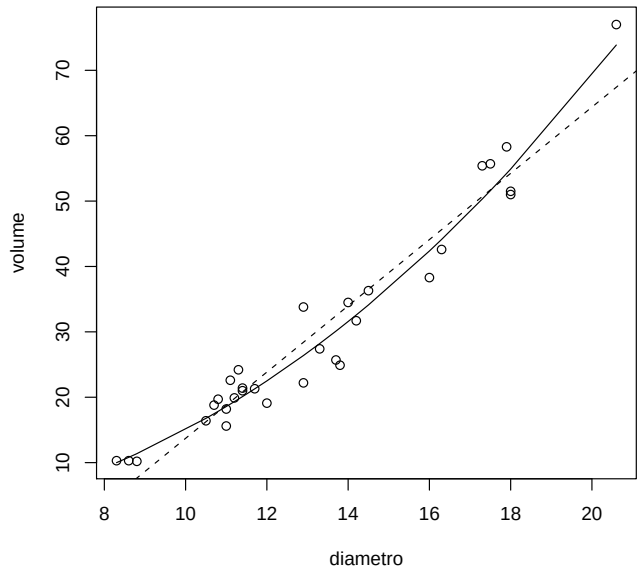


Figura 2.4: Diagramma di dispersione di `diametro` e `volume`, retta di regressione stimata e curva derivante dalla retta di regressione stimata con le variabili in scala logaritmica.

```
> residui1 <- volume - valori.predetti1
```

si nota che la varianza di questi residui è minore rispetto a quella del primo modello utilizzato:

```
> var(residui1)

[1] 10.56006

> var(residui)

[1] 17.47675
```

Per finire, con il comando

```
> detach(Ciliegi)
```

si disattiva il collegamento al *data frame* `Ciliegi`.

Capitolo 3

Distribuzioni e studi di simulazione

3.1 Distribuzione normale

R dispone di alcune funzioni di base per calcolare densità, funzione di ripartizione e quantili per molte distribuzioni di interesse. È inoltre possibile generare realizzazioni pseudo-casuali dalle stesse distribuzioni. Si consideri ad esempio la distribuzione normale standard. sono disponibili 4 funzioni:

- `dnorm(x)` calcola il valore della densità in `x`;
- `pnorm(x)` calcola il valore della ripartizione in `x`;
- `qnorm(p)` calcola il quantile di livello `p`;
- `rnorm(n)` genera un campione da una normale standard di dimensione `n`.

Si può, ad esempio, ottenere il grafico della funzione di ripartizione di una normale standard con:

```
> curve(pnorm, from = -5, to = 5, n = 100)
```

Possono essere trattate anche distribuzioni normali qualsiasi specificandone i parametri. Con il seguente comando si ottiene l'elenco completo degli argomenti della funzione `pnorm( )` con i relativi valori di *default*.

```
> args(pnorm)
```

```
function (q, mean = 0, sd = 1, lower.tail = TRUE, log.p = FALSE)
NULL
```

La Figura 3.1 mostra il grafico della funzione di ripartizione di una normale $N(2, 0.7^2)$, ottenibile mediante il comando:

```
> curve(pnorm(x, mean = 2, sd = 0.7), add = TRUE,
+     col = 2)
```

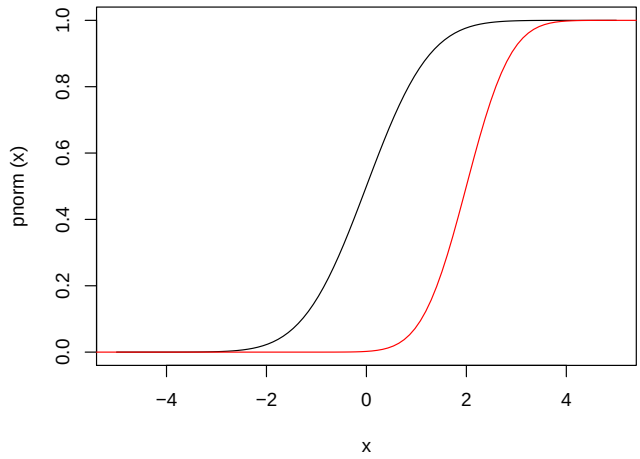


Figura 3.1: Funzione di ripartizione della normale standard (nero) e di una $N(2, 0.7^2)$ (rosso).

Alcune delle distribuzioni disponibili sono elencate nella tabella seguente. Il simbolo - indica che non c'è un valore di *default* per il corrispondente parametro.

R	Distribuzione	Parametri	*Default*
`chisq`	chi-quadrato	df	-
`exp`	esponenziale	rate	1
`f`	F	df1, df2	-, -
`gamma`	Gamma	shape, scale	-, 1
`lnorm`	log-normale	meanlog, sdlog	0, 1
`norm`	normale	mean, sd	0, 1
`t`	t di Student	df	-
`unif`	uniforme	min, max	0, 1

3.2 Adattamento ad una distribuzione normale

Spesso, si desidera valutare graficamente la normalità dei dati. Si generi un campione di numerosità 10 da una distribuzione normale standard

```
> x <- rnorm(10)
> x

 [1]  2.0404811  0.7020486 -0.2228845  1.1850216  1.1582767
 [6]  0.4214277 -0.2222354  0.7796561  1.7340889  0.7437849
```

Si supponga di non sapere che il campione proviene da una popolazione normale e si proceda con alcune analisi grafiche, mostrate in Figura 3.2 e ottenute con:

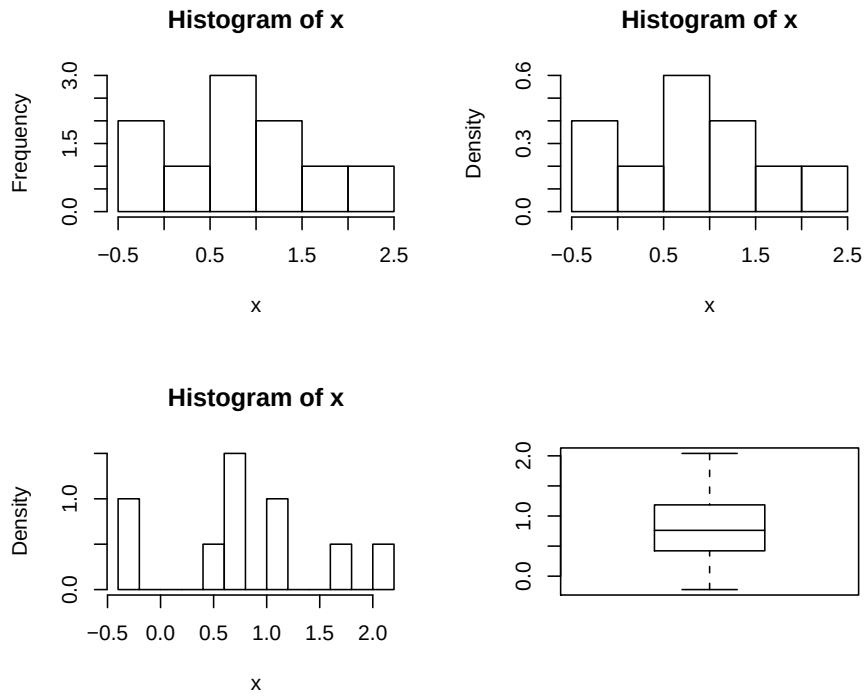


Figura 3.2: Istogrammi di `x` con diverso numero di classi e diagramma a scatola.

```
> par(mfrow = c(2, 2))
> hist(x)
> hist(x, nclass = 8, prob = TRUE)
> hist(x, nclass = 15, prob = TRUE)
> boxplot(x)
```

Si noti l'utilizzo della funzione `par()`, che permette di modificare le opzioni della finestra grafica. In particolare, l'argomento `mfrow=c(r,c)` permette di suddividere la finestra corrente in `r` righe (in questo caso 2) e `c` colonne (in questo caso 2).
L'argomento `prob` della funzione `hist()` posto uguale a `TRUE` fa sì che le aree dei rettangoli che compongono l'istogramma siano uguali alle frequenze relative, anzichè a quelle assolute.
Aumentando la numerosità campionaria, l'istogramma diventa più simile ad una curva di densità normale, come mostrato in Figura 3.3.

```
> par(mfrow = c(1, 2))
> xx <- rnorm(100)
> hist(xx, prob = TRUE, ylim = c(0, 0.5))
> curve(dnorm(x), add = TRUE, col = 2)
> boxplot(xx)
```

Per ottenere il diagramma quantile-quantile o grafico delle probabilità normali (*qq-plot*), si utilizza

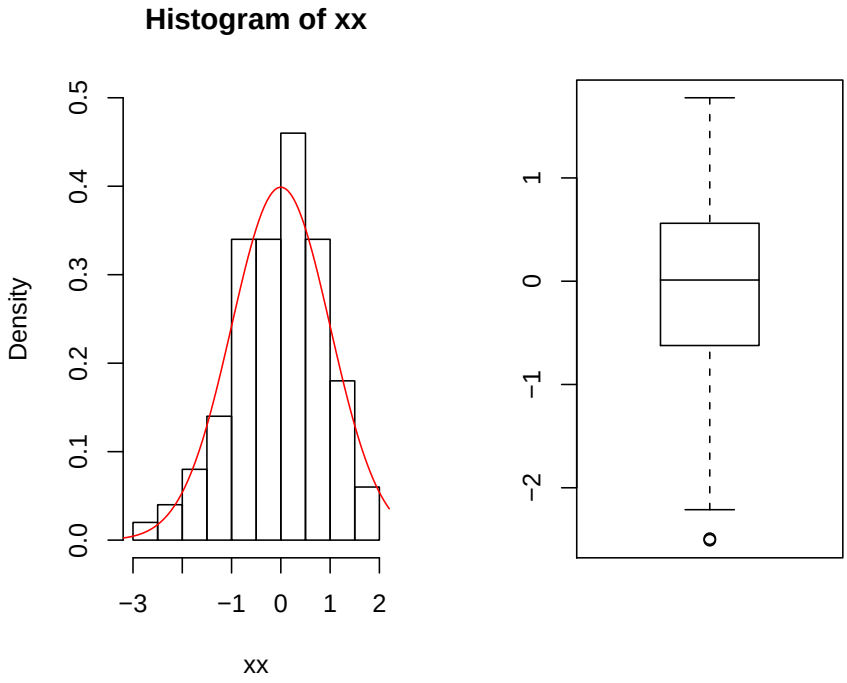


Figura 3.3: Istogramma, con densità normale aggiunta, e diagramma a scatola di `xx`.

```
> qqnorm(xx)
> qqline(xx)
```

Si ottiene il grafico in Figura 3.4.
Esercizio. Si verifichi graficamente la normalità del campione `x`. ◇

Per capire quale può essere l'andamento del *qq-plot* quando i dati non provengono da una distribuzione normale, si considerino dati generati da una distribuzione t_2 e da una distribuzione esponenziale con media 1. I grafici prodotti dai comandi seguenti sono riportati in Figura 3.5.

```
> par(mfrow = c(2, 2))
> y <- rt(100, 2)
> hist(y, prob = TRUE, ylim = c(0, 0.5), nclass = 20)
> curve(dnorm(x), add = TRUE)
> curve(dt(x, 2), add = TRUE, col = 2)
> qqnorm(y)
> qqline(y)
> z <- rexp(100)
> hist(z, prob = TRUE, xlim = c(-4, 4))
> curve(dexp(x), add = TRUE, col = 2, xlim = c(0,
+     4))
> curve(dnorm(x, mean = 1), add = TRUE, col = 3)
```

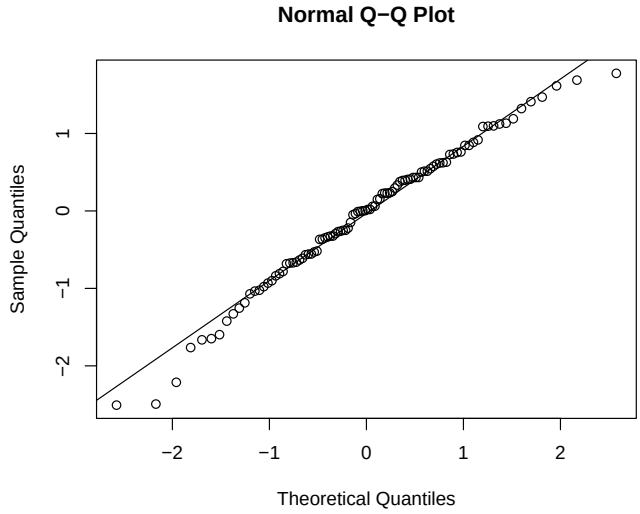


Figura 3.4: `qqnorm` di `xx`.

```
> qqnorm(z)
> qqline(z)
```

La funzione `qqnorm()` non può essere usata per verificare l'adattamento ad una distribuzione diversa dalla normale. In questi casi, occorre utilizzare la funzione `qqplot()`. Per esempio, per la distribuzione esponenziale

```
> qqplot(qexp(ppoints(z)), sort(z))
> abline(0, 1)
```

dove la funzione `ppoints()` fornisce le frequenze cumulate, leggermente modificate, del campione ordinato, e `qxep()` dà i quantili della distribuzione esponenziale. Si ottiene il grafico mostrato in Figura 3.6

3.3 Studio tramite simulazione della distribuzione di $\hat{\beta}_1$ e $\hat{\beta}_2$

Si consideri il modello di regressione lineare semplice normale

$$Y_i = \beta_1 + \beta_2 x_i + \varepsilon_i,$$

con ε_i variabili casuali normali $N(0, \sigma^2)$ indipendenti, $i = 1, \ldots n$. In base alla teoria, si ha

$$\hat{\beta}_2 \sim N(\beta_2, \frac{\sigma^2}{\sum_i (x_i - \bar{x})^2}).$$

È possibile confermare tale risultato tramite uno studio di simulazione. Si consideri il caso particolare in cui $n = 30$, $x_i = i,\ i = 1, \ldots, 30$, $\beta_1 = 5$ $\beta_2 = 3$ e $\sigma^2 = 16$.

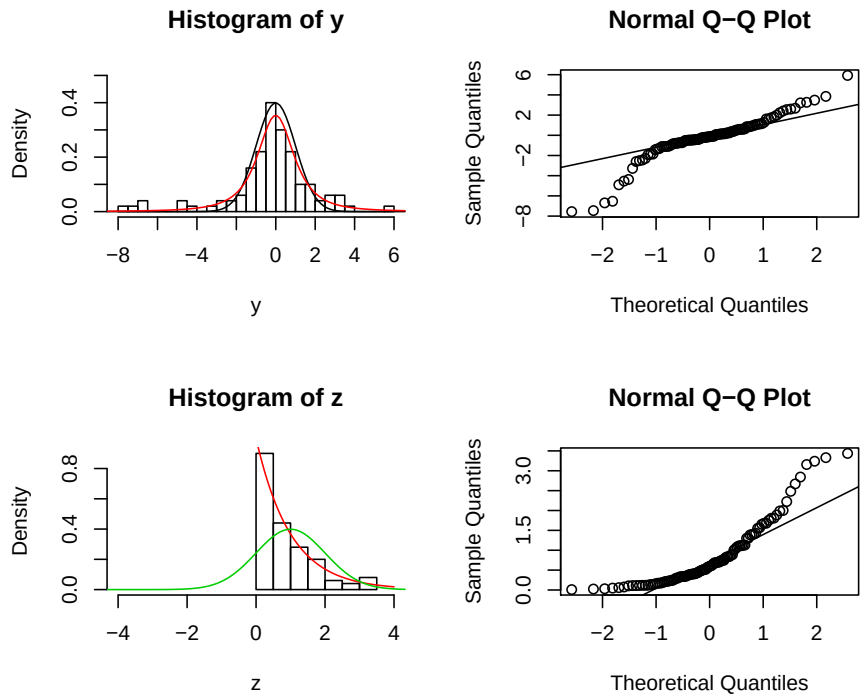


Figura 3.5: Istogramma e *qq-plot* di `y` e `z`.

```
> x <- 1:30
> error <- rnorm(30, mean = 0, sd = 4)
> y <- 5 + 3 * x + error
> plot(x, y)
```

La Figura 3.7 mostra il diagramma di dispersione dei dati simulati.
Si ottengono le stime di $\beta_j,\ j = 1, 2$, con

```
> beta2.hat <- cov(x, y)/var(x)
> beta2.hat
```

```
[1] 3.067723
```

```
> beta1.hat <- mean(y) - beta2.hat * mean(x)
> beta1.hat
```

```
[1] 3.982452
```

e si può confrontare la retta di regressione stimata con la vera retta di regressione (cfr Figura 3.8) mediante i comandi seguenti.

```
> abline(beta1.hat, beta2.hat)
> abline(5, 3, col = 2)
```

Se si aumenta la varianza dell'errore, i punti risultano più dispersi, come mostrato in Figura 3.9.

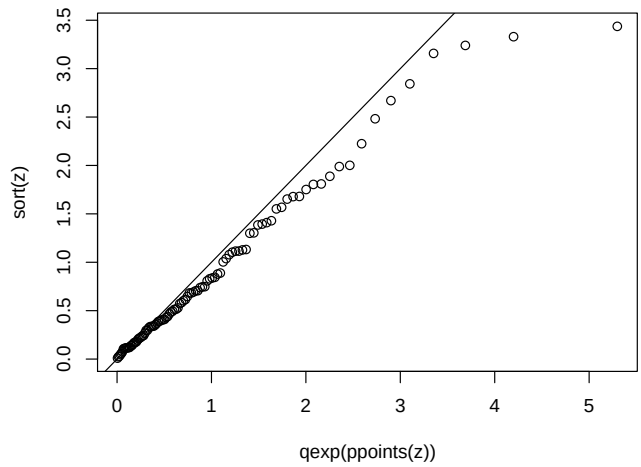


Figura 3.6: *qq-plot* di `z` e la distribuzione esponenziale.

```
> error <- rnorm(30, mean = 0, sd = 10)
> y <- 5 + 3 * x + error
> plot(x, y)
```

Esercizio. Si ripeta l'analisi aumentando e diminuendo σ^2. Si commentino i risultati ottenuti. $\diamond$

Generando campioni diversi dallo stesso modello, si ottengono valori diversi delle stime. Con i comandi seguenti si ottiene un vettore che contiene 1000 realizzazioni della variabile casuale $\hat{\beta}_2 = \hat{\beta}_2(Y)$ quando $(\beta_1, \beta_2, \sigma^2) = (5, 3, 16)$.

```
> beta2.hat.sim <- vector("numeric", length = 1000)
> for (i in 1:1000) {
+     error <- rnorm(30, mean = 0, sd = 4)
+     y <- 5 + 3 * x + error
+     beta2.hat.sim[i] <- cov(y, x)/var(x)
+ }
```

Il primo comando definisce un vettore `beta2.hat.sim` che conterrà le realizzazioni. Si è poi utilizzato il ciclo

```
for (i in 1:1000) {  operazioni  }
```

L'indice `i` assume valori da 1 a 1000 e segue le iterazioni. A ciascuna iterazione, vengono eseguite le operazioni all'interno delle parentesi graffe.
Si può valutare graficamente la distribuzione di $\hat{\beta}_2$, come mostrato in Figura 3.10.

```
> par(mfrow = c(1, 2))
> hist(beta2.hat.sim, prob = TRUE)
> boxplot(beta2.hat.sim)
```

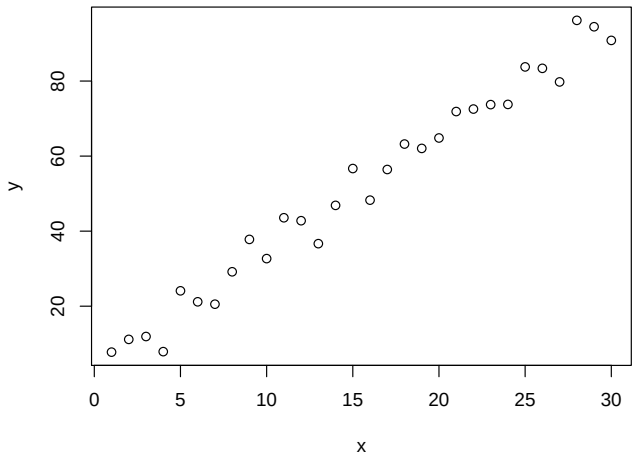


Figura 3.7: Diagramma di dispersione di `x` e `y`.

In base alla teoria, si ha $V(\hat{\beta}_2) = \frac{\sigma^2}{\sum_i (x_i - \bar{x})^2}$, che risulta

```
> var.beta2.hat <- 16/sum((x - mean(x))^2)
> var.beta2.hat

[1] 0.007119021
```

per cui la distribuzione esatta è

$$\hat{\beta}_2 \sim N(3, 0.00712).$$

Si può confrontare la distribuzione empirica con quella esatta (cfr Figura 3.11).

```
> mean(beta2.hat.sim)

[1] 2.998246

> var(beta2.hat.sim)

[1] 0.006873917

> hist(beta2.hat.sim, prob = TRUE)
> curve(dnorm(x, mean = 3, sd = sqrt(var.beta2.hat)),
+     add = TRUE)
```

Esercizio. Si ottenga la distribuzione simulata di $\hat{\beta}_1$. ◇

Si calcoli ora l'intervallo di confidenza con livello esatto 0.95 per β_2, $\hat{\beta}_2 \pm t_{n-2;0.975} \sqrt{\hat{V}(\hat{\beta}_2)}$.

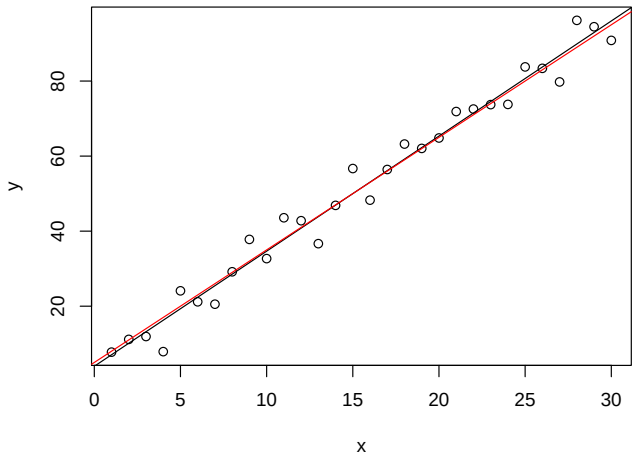


Figura 3.8: Diagramma di dispersione di `x` e `y`, retta di regressione stimata (nero) e "vera" retta di regressione (rosso).

```
> n <- length(x)
> beta2.hat <- cov(x, y)/var(x)
> beta1.hat <- mean(y) - beta2.hat * mean(x)
> residui <- y - beta1.hat - beta2.hat * x
> s2 <- sum(residui^2)/(n - 2)
> var.beta2.hat <- s2/sum((x - mean(x))^2)
> beta2.lower <- beta2.hat - qt(0.975, n - 2) *
+     sqrt(var.beta2.hat)
> beta2.upper <- beta2.hat + qt(0.975, n - 2) *
+     sqrt(var.beta2.hat)
> beta2.lower

[1] 2.980454

> beta2.upper

[1] 3.206911
```

Secondo la teoria, replicando l'esperimento un numero elevato di volte, circa il 95% degli intervalli di confidenza osservati dovrebbe contenere il vero valore del parametro, $\beta_2 = 3$. Si può ricorrere ad uno studio di simulazione anche per valutare il livello di copertura *effettivo* di un intervallo di confidenza con livello *nominale* 0.95.

Si crei un file di testo `simul.R` (usando un qualsiasi editor di testo) con le istruzioni necessarie per generare 1000 campioni, su cui calcolare gli intervalli di confidenza, e memorizzare gli estremi nella matrice con 1000 righe e 2 colonne di nome `beta2.ic`.

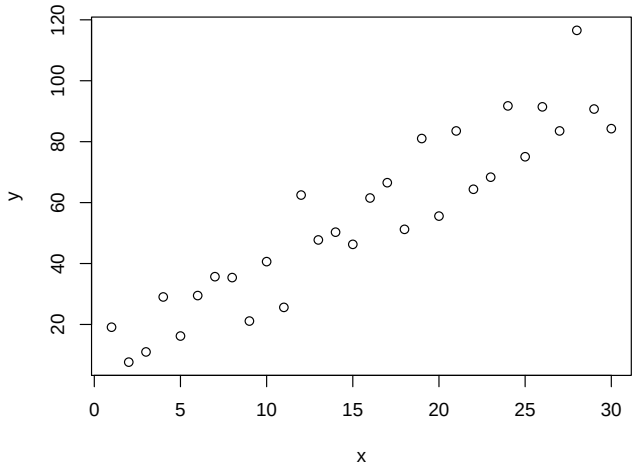


Figura 3.9: Diagramma di dispersione di `x` e `y`, con varianza dell'errore pari a 100.

```
> beta2.ic <- matrix(NA, ncol = 2, nrow = 1000)
> for (i in 1:1000) {
+     error <- rnorm(30, mean = 0, sd = 4)
+     y <- 5 + 3 * x + error
+     beta2.hat <- cov(x, y)/var(x)
+     beta1.hat <- mean(y) - beta2.hat * mean(x)
+     residui <- y - beta1.hat - beta2.hat * x
+     s2 <- sum(residui^2)/(n - 2)
+     var.beta2.hat <- s2/sum((x - mean(x))^2)
+     beta2.lower <- beta2.hat - qt(0.975, n - 2) *
+         sqrt(var.beta2.hat)
+     beta2.upper <- beta2.hat + qt(0.975, n - 2) *
+         sqrt(var.beta2.hat)
+     beta2.ic[i, 1] <- beta2.lower
+     beta2.ic[i, 2] <- beta2.upper
+ }
```

I comandi contenuti nel file `simul.R` possono essere eseguiti in `R` con

```
> source("simul.R")
```

oppure facendo copia ed incolla.
Una stima del livello di copertura effettivo è dato dalla proportzione di intervalli che contengono il vero valore di β_2.

```
> mean((beta2.ic[, 1] <= 3) & (beta2.ic[, 2] >=
+     3))
```

```
[1] 0.951
```

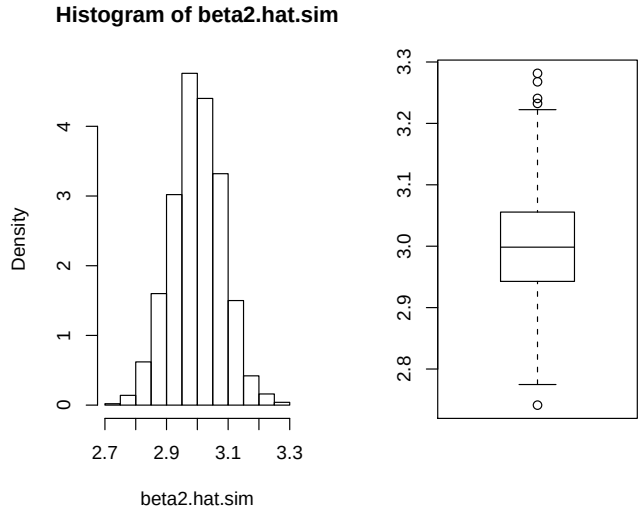


Figura 3.10: Istogramma e diagramma a scatola delle stime di β_2 ottenute da 1000 campioni simulati.

Il valore è prossimo al livello di copertura nominale 0.95.

Si può fare un grafico di alcuni di questi intervalli di confidenza. Ad esempio, per i primi 100 intervalli simulati si ha:

```
> plot(1:100, beta2.ic[1:100, 1], ylim = c(2.5,
+     3.5), ylab = "beta2")
> points(1:100, beta2.ic[1:100, 2])
> segments(1:100, beta2.ic[1:100, 2], 1:100, beta2.ic[1:100,
+     1])
> abline(h = 3, col = 2)
```

Esercizio. Si valuti il livello di copertura effettivo dell'intervallo di confidenza con livello nominale 0.9 per il coefficiente β_1. ◇

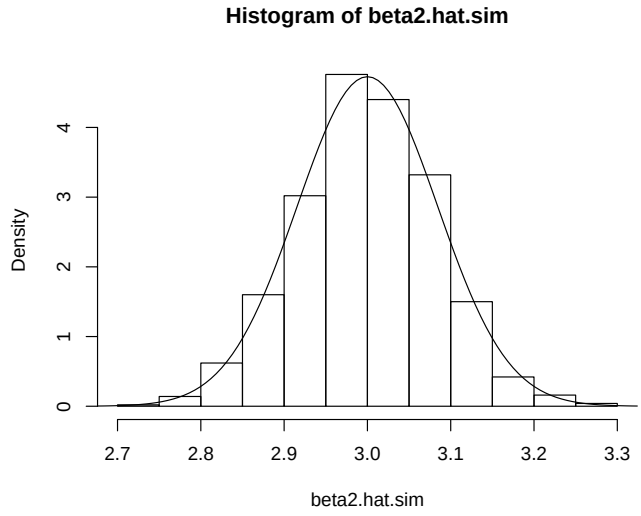


Figura 3.11: Istogramma delle stime di β_2 ottenute da 1000 campioni simulati e distribuzione esatta dello stimatore $\hat{\beta}_2$.

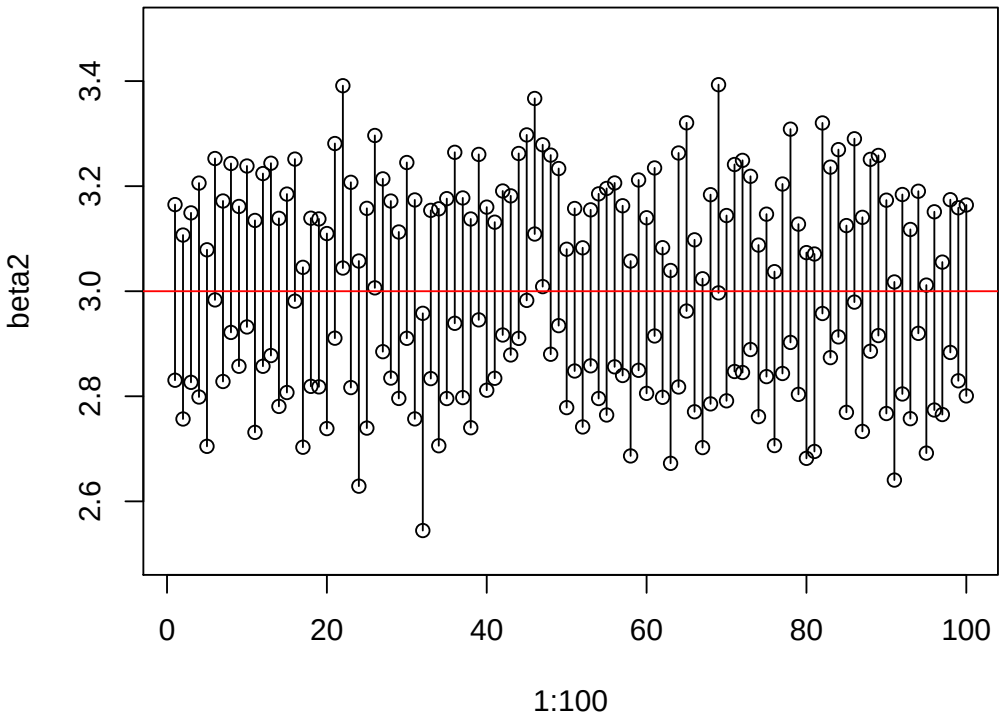


Figura 3.12: 100 intervalli di confidenza da campioni simulati e vero valore del parametro (linea rossa orizzontale).

Capitolo 4

Modello di regressione lineare semplice: la funzione `lm()`

4.1 Analisi dei dati CHERRY.DAT

Si riconsiderino i dati contenuti nel file `cherry.dat`, riferiti a misurazioni rilevate su 31 alberi di ciliegio.

```
> Ciliegi <- read.table("cherry.dat", col.names = c("diametro",
+     "altezza", "volume"))
> attach(Ciliegi)
```

Si consideri il modello già utilizzato nel Paragrafo 2.1 per spiegare la relazione tra volume e diametro, ovvero il modello che assume che $y_1, \ldots, y_{31}$ siano realizzazioni di variabili casuali Y_i dove

$$Y_i = \beta_1 + \beta_2 x_i + \varepsilon_i \,, \tag{4.1}$$

con ε_i variabili casuali $N(0, \sigma^2)$ indipendenti, $i = 1, \ldots, 31$.

Per stimare il modello, si può utilizzare la funzione `lm()`. La funzione prevede la specificazione di una formula che definisce il modello da stimare.

```
> ciliegi.lm <- lm(volume ~ diametro)
```

Si noti la sintassi della formula, `volume~diametro` . A sinistra di ~ vi è la variabile risposta, mentre a destra la variabile esplicativa. L'intercetta β_1 è automaticamente inclusa.
La funzione `lm()` produce un oggetto, memorizzato con il nome `ciliegi.lm`, di classe `lm`. In `R`, un oggetto è una struttura più complicata di un vettore o di una matrice. Si tratta di una lista di elementi di diverso tipo (vettori, matrici,...). La classe di un oggetto determina il modo in cui alcune funzioni elementari agiscono su di esso e quindi anche il risultato che forniscono. Un esempio importante è dato dalla funzione `summary()` che, applicata ad un oggetto di classe `lm`, produce il seguente risultato.

```
> summary(ciliegi.lm)
```

```
Call:
lm(formula = volume ~ diametro)

Residuals:
    Min      1Q  Median      3Q     Max
-8.0654 -3.1067  0.1520  3.4948  9.5868

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -36.9435     3.3651  -10.98 7.62e-12 ***
diametro      5.0659     0.2474   20.48  < 2e-16 ***
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

Residual standard error: 4.252 on 29 degrees of freedom
Multiple R-Squared: 0.9353,         Adjusted R-squared: 0.9331
F-statistic: 419.4 on 1 and 29 DF,  p-value: < 2.2e-16
```

È importante capire e saper interpretare questo riassunto dell'analisi. Innanzi tutto, viene ricordato, nella sintassi di `R`, il modello stimato (`Call`). Vengono poi fornite alcune statistiche riassuntive relative ai residui del modello stimato (`Residuals`): il minimo, il massimo e i tre quartili.
La parte più importante del risultato è la tabella relativa ai coefficienti del modello (`Coefficients`), in cui ogni riga corrisponde ad un coefficiente: `(Intercept)` rappresenta β_1, mentre il nome della variabile (in questo caso `diametro`) rappresenta il relativo coefficiente (β_2). Per ciascuna riga j, $j = 1, 2$, la tabella riporta la stima $\hat{\beta}_j$ (`Estimate`), la radice della stima non distorta della varianza dello stimatore $\sqrt{\hat{\mathrm{V}}(\hat{\beta}_j)}$ (`Std. Error`), il valore osservato t_j^{oss} del test $t_j = \hat{\beta}_j / \sqrt{\hat{\mathrm{V}}(\hat{\beta}_j)}$ per la verifica dell'ipotesi $H_0 : \beta_j = 0$ vs $H_1 : \beta_j \neq 0$ (`t value`) ed infine il livello di significatività osservato $\alpha^{\mathrm{oss}} = 2\ \mathrm{Pr}_{H_0}(t_j > |t_j^{\mathrm{oss}}|)$, indicato con `Pr(>|t|)`. Per facilitare l'interpretazione dei livelli di significatività osservati, a fianco di ognuno viene apposto un simbolo. La legenda per questi simboli si trova in calce alla tabella dei coefficienti. In particolare, `***` indica che $\alpha^{\mathrm{oss}} < 0.001$ (e cioè una marcata significatività), `**` indica che $0.001 < \alpha^{\mathrm{oss}} < 0.01$ (comunque una forte significatività), e così via fino ad arrivare all'assenza di simboli, che si ha quando $\alpha^{\mathrm{oss}} > 0.10$ (non si rifiuta l'ipotesi di nullità del coefficiente).
La voce `Residual standard error` riporta la radice quadrata della stima corretta della varianza σ^2 del termine d'errore, ovvero s, assieme ai relativi gradi di libertà $(n-2)$.
Il coefficiente di determinazione, R^2, e una sua versione corretta per il numero di variabili presenti nel modello, R^2_{adj}, sono riportati rispettivamente nelle voci `Multiple R-Squared` e `Adjusted R-Squared`. La loro espressione è:

$$\begin{aligned} R^2 &= 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}\,, \\ R^2_{adj} &= R^2 - \frac{p-1}{n-p}(1 - R^2)\,. \end{aligned}$$

Infine, è anche riportato il valore osservato del test F per la verifica dell'ipotesi di nullità di tutti i coefficienti di regressione tranne l'intercetta. In questo modello, tale test verifica l'ipotesi di nullità del coefficiente angolare, ovvero l'ipotesi $H_0 : \beta_2 = 0$ vs $H_1 : \beta_j \neq 0$ Sotto l'ipotesi nulla, la statistica F ha distribuzione $F_{1,n-2} \sim t^2_{n-2}$ e vale l'identità $F = t^2_2$. Il livello di significatività osservato α^{oss} (`p-value`) indica una forte evidenza contro l'ipotesi nulla ed è, ovviamente, coincidente con quello relativo al test t_2.

4.2 Analisi dei dati BRAINBOD.DAT

Si consideri il file `brainbod.dat` che contiene i seguenti dati sul peso del corpo (in kg) e peso del cervello (in g) di 15 mammiferi terrestri.

species	bodywt	brainwt
afeleph	6654.00	5712.00
cow	465.00	423.00
donkey	187.00	419.00
man	62.00	1320.00
graywolf	36.33	119.50
redfox	4.24	50.40
narmadillo	3.50	10.80
echidna	3.00	25.00
phalanger	1.62	11.40
guineapig	1.04	5.50
eurhedghog	.79	3.50
chinchilla	.43	4.00
ghamster	.12	1.00
snmole	.06	1.00
lbbat	.01	.25

Dopo aver cancellato lo spazio di lavoro, si acquisiscono i dati

```
> rm(list = ls())
> Brainbod <- read.table("brainbod.dat", header = TRUE)
> attach(Brainbod)
```

La Figura 4.1 mostra una analisi grafica preliminare dei dati, ottenuta tramite i comandi che seguono.

```
> hist(bodywt)

> boxplot(bodywt)

> hist(brainwt)

> boxplot(brainwt)
```

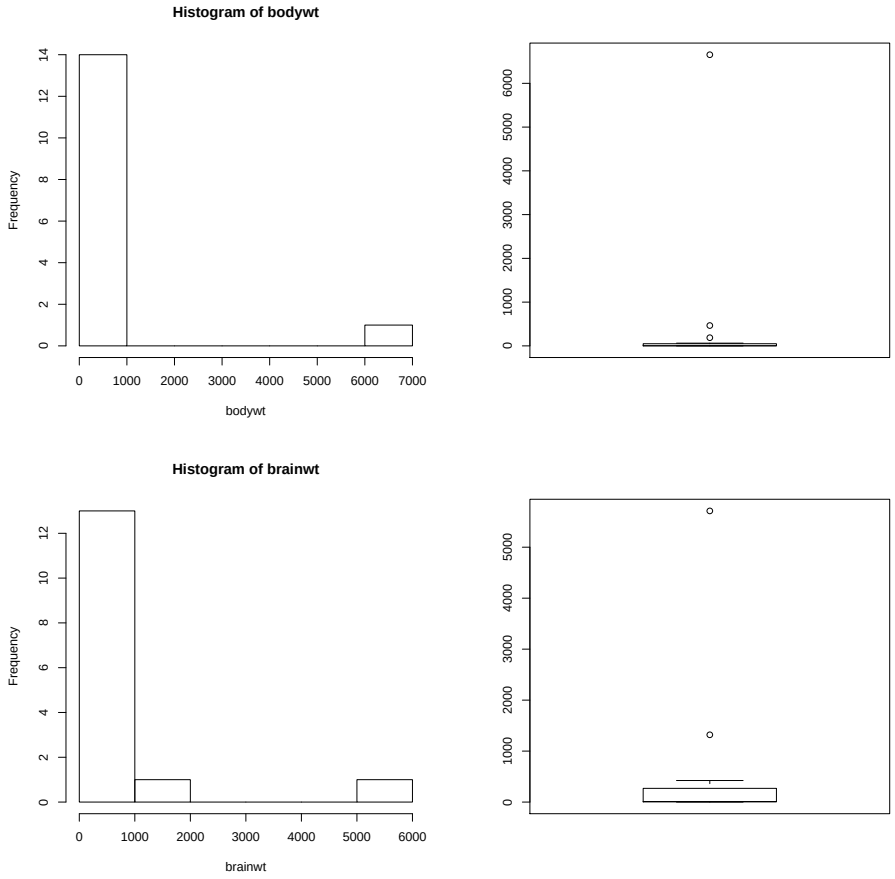


Figura 4.1: Istogramma e diagramma a scatola di `bodywt` (rispettivamente in alto a sinistra e in alto a destra) e `brainwt` (rispettivamente in basso a sinistra e in basso a destra).

Siano (x_i, y_i), $i = 1, \ldots, 15$, i dati relativi alle due variabili peso del corpo e peso del cervello. Il diagramma di dispersione delle coppie (x_i, y_i), $i = 1, \ldots, 15$, riportato in Figura 4.2, è ottenuto mediante il seguente comando.

```
> plot(bodywt, brainwt)
```

Per identificare le unità statistiche a cui corrispondono i punti nel diagramma di dispersione, si può utilizzare il comando interattivo:

```
> identify(bodywt, brainwt, species)
```

I grafici nelle Figure 4.1 e 4.2 sono difficili da interpretare, perchè il dato relativo all'elefante è molto più elevato di quello relativo alle restanti unità. Si può provare allora ad esplorare i dati su scala logaritmica, come mostrato in Figura 4.3 e in Figura 4.4.

```
> par(mfrow = c(2, 2))
> hist(log(bodywt))
```

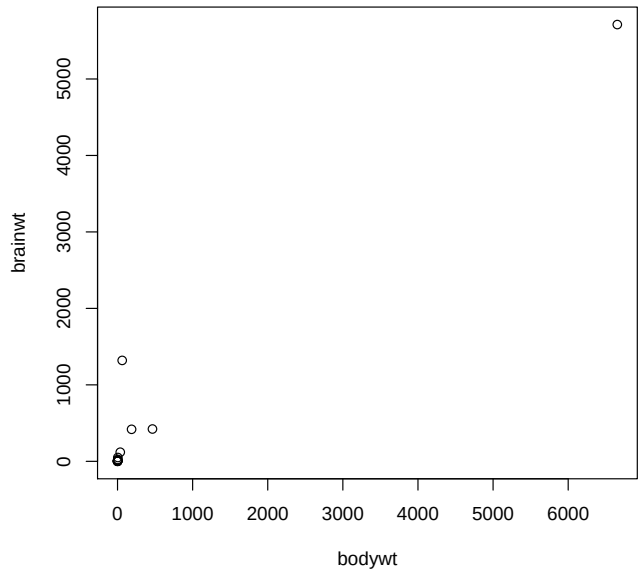


Figura 4.2: Diagramma di dispersione di `bodywt` e `brainwt`.

```
> boxplot(log(bodywt))
> hist(log(brainwt))
> boxplot(log(brainwt))
```

```
> plot(log(bodywt), log(brainwt))
```

I grafici ottenuti, riportati nelle Figure 4.3 e 4.4 appaiono più interpretabili. Si nota un allineamento tendenziale dei punti.

Esercizio. Si stimino, utilizzando la funzione `lm()`, i coefficienti del modello di regressione lineare semplice normale

$$\log(Y_i) = \beta_1 + \beta_2 z_i + \varepsilon_i \,,$$

con $z_i = \log(x_i)$ e ε_i variabili casuali normali $N(0, \sigma^2)$ indipendenti, $i = 1, \ldots, 15$. Si calcolino inoltre gli intervalli di confidenza con livello 0.99 per β_1 e per β_2. $\diamond$

Dall'esercizio precedente si ha $\hat{\beta}_1$ =2.048 e $\hat{\beta}_2$ =0.782. Si definiscano due oggetti `R` di nome `beta1.hat` e `beta2.hat` contenenti i valori di $\hat{\beta}_1$ e $\hat{\beta}_2$. Si costruiscano poi i valori predetti ed i residui.

```
> brainbod.fit = beta1.hat + beta2.hat * log(bodywt)
> brainbod.res = log(brainwt) - brainbod.fit
```

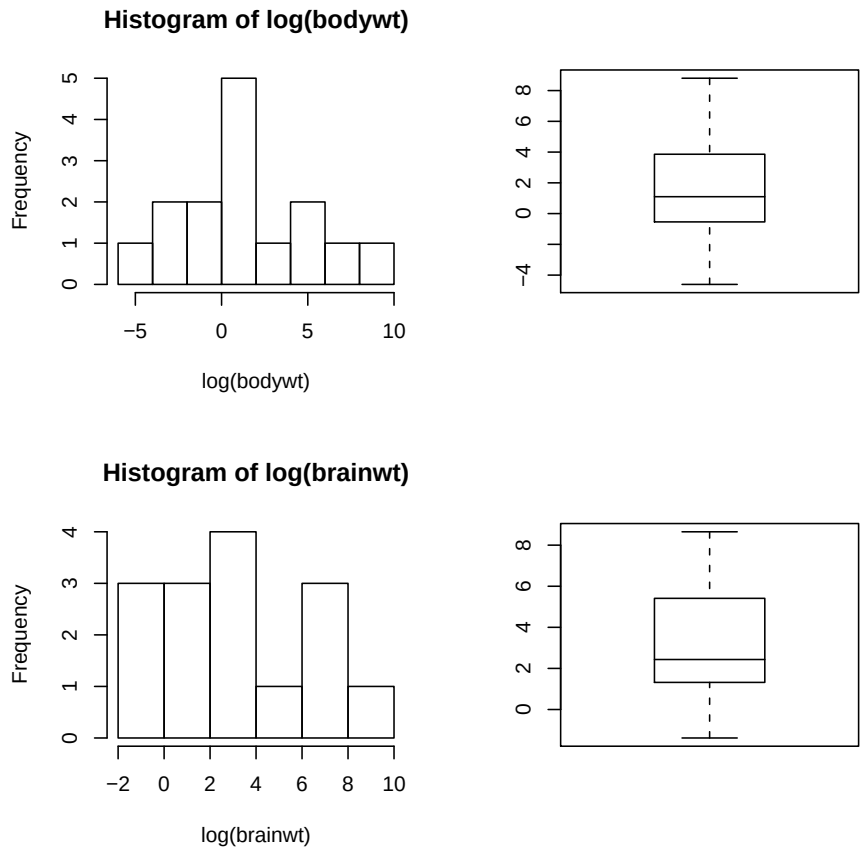


Figura 4.3: Istogramma e diagramma a scatola di `bodywt` (rispettivamente in alto a sinistra e in alto a destra) e `brainwt` (rispettivamente in basso a sinistra e in basso a destra), su scala logaritmica.

Si possono verificare numericamente le proprietà algebriche dei residui. In particolare

```
> sum(brainbod.res)

[1] 3.996803e-15

> sum(brainbod.res * log(bodywt))

[1] 4.13428e-15

> sum(brainbod.res * brainbod.fit)

[1] 1.081427e-14
```

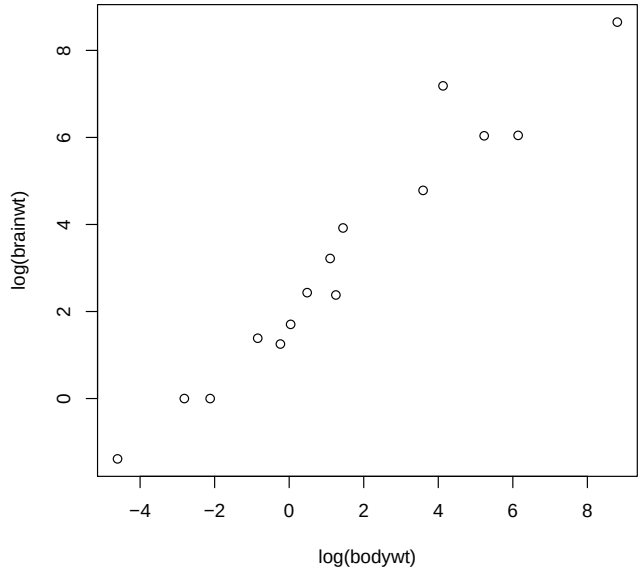


Figura 4.4: Diagramma di dispersione di `bodywt` e `brainwt`, su scala logaritmica.

Capitolo 5

Costruzione del modello e analisi dei residui

5.1 Analisi dei dati CEMENT.DAT

I dati 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 essiccazione. Nello studio è stata misurata la resistenza alla tensione (in kg/cm^2) di lotti di cemento sottoposti a diversi tempi di essiccazione (in giorni). Si vuole studiare la relazione tra la resistenza alla tensione e il tempo di essiccazione. In questo caso il tempo, $t = (t_1, \ldots, t_n)$, è la variabile esplicativa e la resistenza, $y = (y_1, \ldots, y_n)$, è la variabile risposta.

```
> rm(list = ls())
> Cem <- read.table("cement.dat", col.names = c("tempo",
+     "resist"))
> attach(Cem)
```

È utile iniziare con un'analisi esplorativa dei dati, per esempio osservando il diagramma di dispersione tra la variabile risposta e la variabile esplicativa.

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

La Figura 5.1 indica chiaramente una relazione non lineare. Quindi, un modello che ipotizza che $y_i,\ i = 1, \ldots, n$, siano realizzazioni di variabili casuali indipendenti $Y_i \sim N(\beta_1 + \beta_2 t_i, \sigma^2)$ non sembra appropriato. Conviene allora cercare qualche trasformazione delle variabili che linearizzi la relazione.
Generalmente, si preferisce trasformare le variabili esplicative. Si possono provare diverse trasformazioni della variabile t. La Figura 5.2 mostra l'effetto di 4 trasformazioni sul diagramma di dispersione.

```
> 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))
```

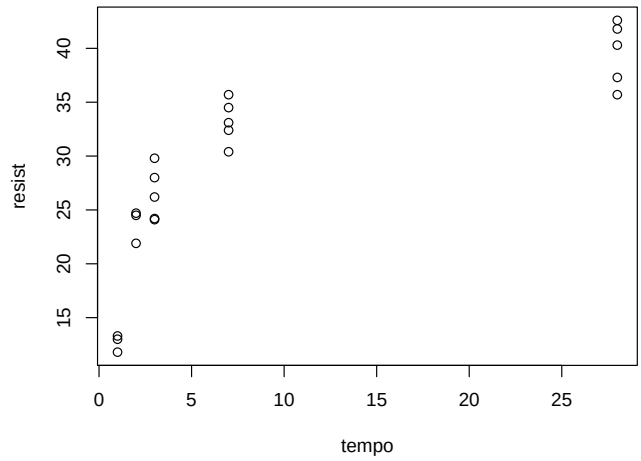


Figura 5.1: Diagramma di dispersione di `tempo` e `resist`.

Le prime tre trasformazioni forniscono una soddisfacente linearizzazione della relazione, in particolare la terza: $x_i = 1/\sqrt{t_i},\ i = 1, \ldots, n$. Si applichi quindi questa trasformazione.

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

e si consideri il seguente modello di regressione lineare semplice normale:

$$Y_i = \beta_1 + \beta_2 x_i + \varepsilon_i\,,$$

con x_i rappresentante la temperatura trasformata e ε_i variabili casuali normali $N(0, \sigma^2)$ indipendenti, $i = 1, \ldots, n$.
Si stimi il modello tramite la funzione `lm()`.

```
> cem.lm <- lm(resist ~ x)
```

Come già anticipato, l'oggetto `cem`, di classe `lm`, contiene diverse quantità.

```
> names(cem.lm)
```

```
 [1] "coefficients"  "residuals"     "effects"
 [4] "rank"          "fitted.values" "assign"
 [7] "qr"            "df.residual"   "xlevels"
[10] "call"          "terms"         "model"
```

Si può accedere agli elementi della lista `cem` attraverso l'operatore `$`. Ad esempio, il comando `cem.lm$coefficients` restituisce il vettore $\hat{\beta} = (\hat{\beta}_1, \hat{\beta}_2)^\top$, il comando `cem.lm$fitted.values` restituisce il vettore $\hat{y} = (\hat{y}_1, \ldots, \hat{y}_n)^\top$, con $\hat{y}_i = \hat{\beta}_1 + \hat{\beta}_2 x_i,\ i = 1, \ldots, n$, il comando `cem.lm$residuals` restituisce il vettore $e = (e_1, \ldots, e_n)^\top$, con $e_i = y_i - \hat{y}_i,\ i = 1, \ldots, n$. Vista l'importanza delle quantità sopra ricordate, esistono alcune funzioni che procedono alla loro estrazione dall'oggetto `cem.lm` evitando il ricorso all'operatore `$`.

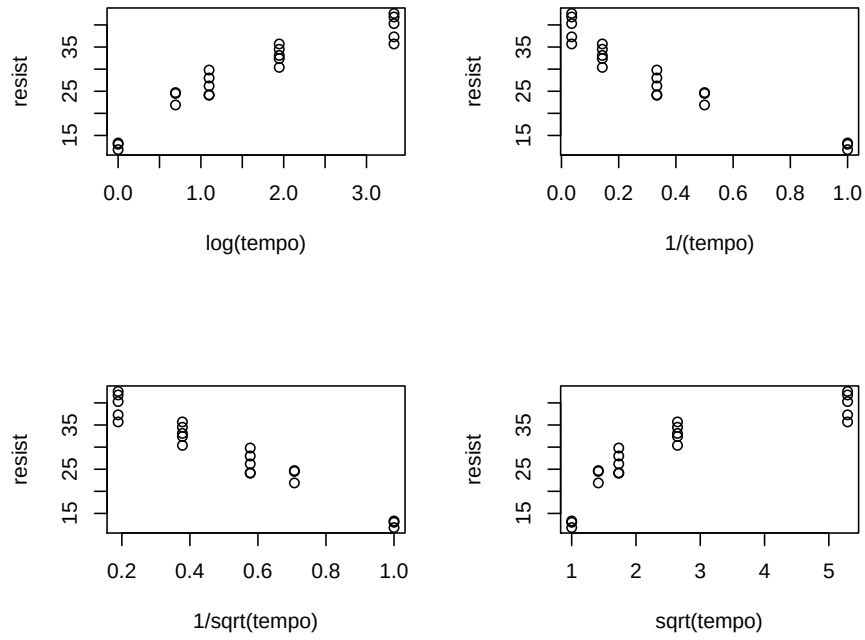


Figura 5.2: Diagramma di dispersione di diverse trasformazioni di `tempo` e `resist`.

```
> cem.coeff <- coef(cem.lm)
> cem.fit <- fitted(cem.lm)
> cem.res <- resid(cem.lm)
```

I residui non sono omoschedastici; infatti $V(e_i) = \sigma^2(1 - h_i)$, dove

$$h_i = 1/n + (x_i - \bar{x})^2 / \sum_{i=1}^{n} (x_i - \bar{x})^2$$

, $i = 1, \ldots, n$. Si possono però utilizzare i residui standardizzati $e_i^* = e_i/(s\sqrt{1 - h_i})$, che hanno distribuzione approssimata normale standard. I residui standardizzati si ottengono utilizzando la funzione `rstandard()`.

```
> cem.rstand <- rstandard(cem.lm)
```

Alcuni grafici che sono utili per verificare la linearità della relazione, l'omoschedasticità degli errori e l'indipendenza degli errori sono i diagrammi di dispersione dei seguenti punti ($i = 1, \ldots, n$)

(1) $(\hat{y}_i, e_i^*)$;

(2) $(\hat{y}_i, y_i)$;

(3) (x_i, e_i^*).

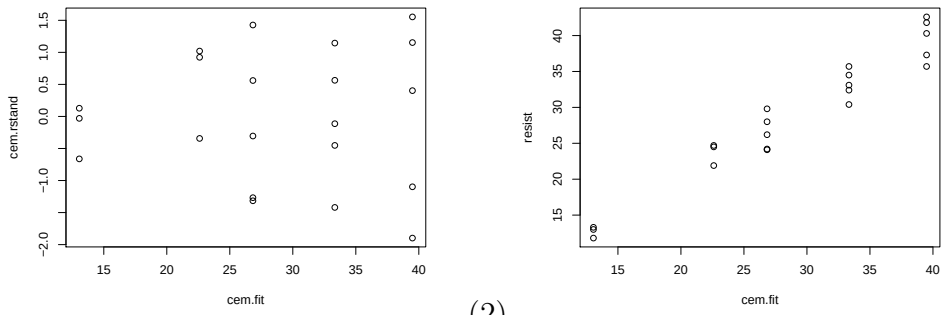
(4) (i, e_i^*), utile soprattutto se le osservazioni sono in ordine temporale;

Il diagramma di dispersioine dei punti (1), ossia dei residui rispetto ai valori stimati, ottenuto mediante il comando:

```
> plot(cem.fit, cem.rstand)
```

e riportato in Figura **??** 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.
Nella stessa figura, lo stesso andamento, anche se meno evidente, è mostrato anche dal grafico (2) dei valori osservati rispetto valori stimati, ottenibile mediante il comando:

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



(1) (2)

Figura 5.3: Grafico dei residui standardizzati verso i valori stimati (sinistra) e di `resist` verso i valori stimati (destra).

Per esplorare la normalità dei residui standardizzati, si possono usare i comandi che seguono, che producono la Figura 5.4.

```
> par(mfrow = c(1, 3))
> hist(cem.rstand, freq = FALSE)
> curve(dnorm(x), add = TRUE)
> boxplot(cem.rstand)
> qqnorm(cem.rstand, xlim = c(-2, 2), ylim = c(-2,
+     2))
> qqline(cem.rstand)
```

Complessivamente, la normalità appare soddisfacente considerando la bassa numerosità del campione, seppure i grafici quantile-quantile evidenzino sulla coda destra lievi deviazioni dei residui osservati rispetto al comportamento atteso. Concludendo, il modello interpola i dati abbastanza bene; esso risulta peraltro un po' carente per quanto riguarda la omoschedasticità del termine di errore.

Esercizio. Produrre diagramma di dispersione dei punti (i, e_i^*) e commentarlo. Spesso, in pratica, anziché utilizzare i residui standardizzati si utilizzano i residui non standardizzati e_i. Si ripetano le analisi grafiche precedenti con i residui non standardizzati e si commentino eventuali somiglianze o differenze. ◇

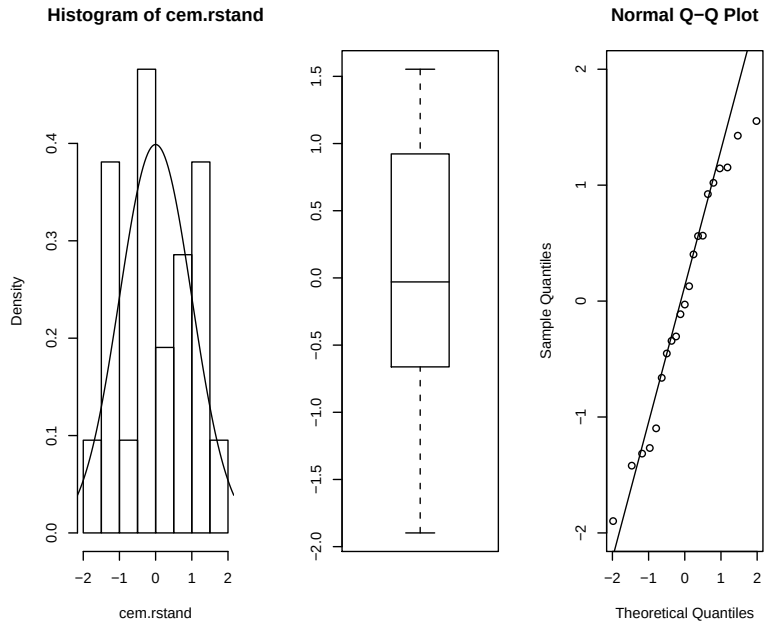


Figura 5.4: Istogramma (sinistra), diagramma a scatola (centro) e *qq-plot* (destra) dei residui standardizzati.

5.2 Analisi dei dati WINDMILL.DAT

Si è interessati alla relazione che intercorre tra velocità del vento, x (in miglia orarie), e corrente generata da una turbina eolica, y (corrente diretta). A tal fine, sono stati raccolti alcuni dati, memorizzati nel file `windmill.dat`.

```
> Vento <- read.table("windmill.dat", header = TRUE)
> attach(Vento)
```

Per esplorare graficamente la relazione esistente tra velocità del vento e corrente generata è utile partire da un diagramma di dispersione, mostrato in Figura 5.5.

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

Il grafico mostra una evidente relazione tra le due variabili. Pare ragionevole provare ad ipotizzare l'esistenza di un legame lineare tra le variabili. Si assume quindi il modello

$$Y_i = \beta_1 + \beta_2 x_i + \varepsilon_i \,,$$

con ε_i, $i = 1, \ldots, n$, variabili casuali normali $N(0, \sigma^2)$ indipendenti, stimabile attraverso la funzione `lm()`.

```
> vento.lm <- lm(dc ~ wind)
> summary(vento.lm)
```

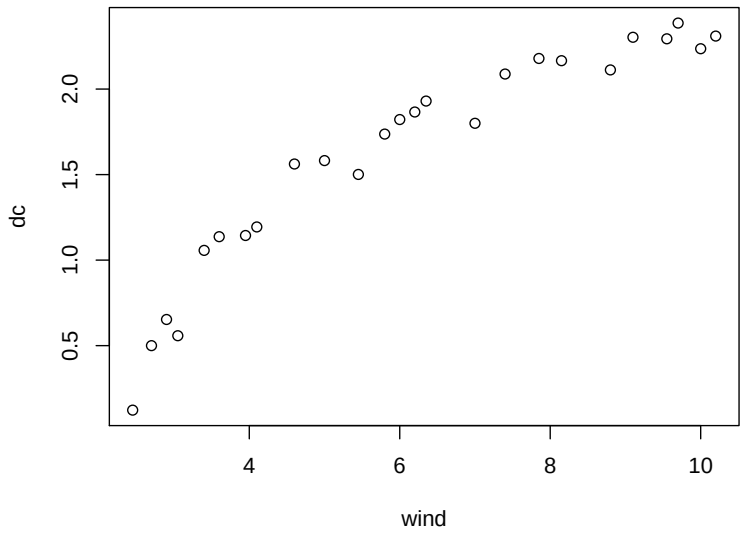


Figura 5.5: Diagramma di dispersione di `wind` e `dc`.

```
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, si può valutare la significatività dei coefficienti e passare poi all'analisi dei residui.
Si noti anche qui l'identità dei livelli di significatività osservati del test t_2 sulla nullità di β_2 e del test F, dovuta al fatto che $F = t_2^2$.

```
> 12.659^2

[1] 160.2503
```

Si può procedere ora con l'analisi grafica dei residui, mostrata in Figura 5.6.

```
> par(mfrow = c(1, 2))
> vento.rstand <- rstandard(vento.lm)
> vento.fit <- fitted(vento.lm)
> plot(vento.fit, vento.rstand)
> plot(wind, vento.rstand)
```

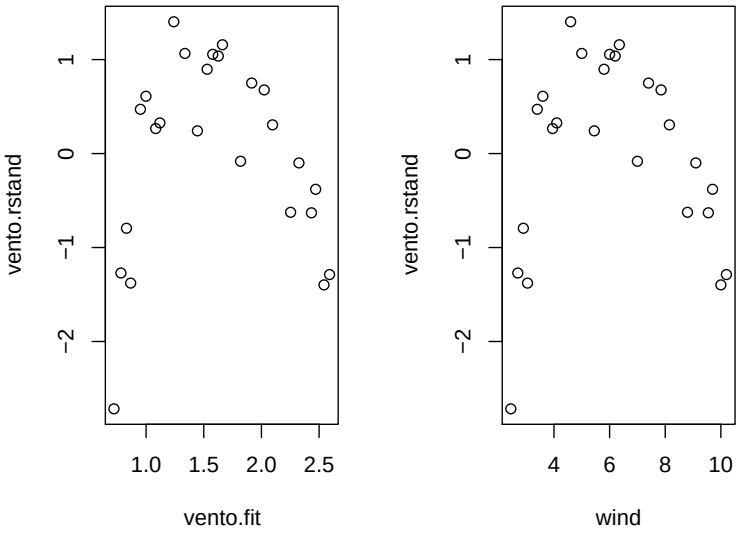


Figura 5.6: Grafico dei residui standardizzati, rispettivamente rispetto ai valori stimati dal modello e rispetto a `wind`.

I grafici dei residui standardizzati, e_i^*, rispetto ai valori stimati, $\hat{y}_i$, e rispetto alla variabile esplicativa, x_i, indicano un chiaro andamento parabolico dei residui standardizzati. In particolare, il grafico (x_i, e_i^*) mostra che, in corrispondenza di valori bassi e alti della velocità del vento, il modello sistematicamente sovrastima il valore della corrente generata ($e_i^* < 0$), mentre in corrispondenza di valori centrali della velocità, il modello sottostima la corrente generata ($e_i^* > 0$). Questo andamento suggerisce che il modello non coglie in maniera appropriata la dipendenza della variabile risposta dalla esplicativa.
Inoltre, il grafico quantile–quantile, ottenuto con i comandi

```
> qqnorm(vento.rstand)
> qqline(vento.rstand)
```

e mostrato in Figura 5.7, evidenzia qualche scostamento dalla normalità per i residui di segno positivo. Da questo si potrebbe desumere che la distribuzione dei residui sia asimmetrica. In conclusione, l'analisi dei residui non appare soddisfacente, nonostante il test di nullità di β_2 indichi un effetto statisticamente significativo della variabile esplicativa.

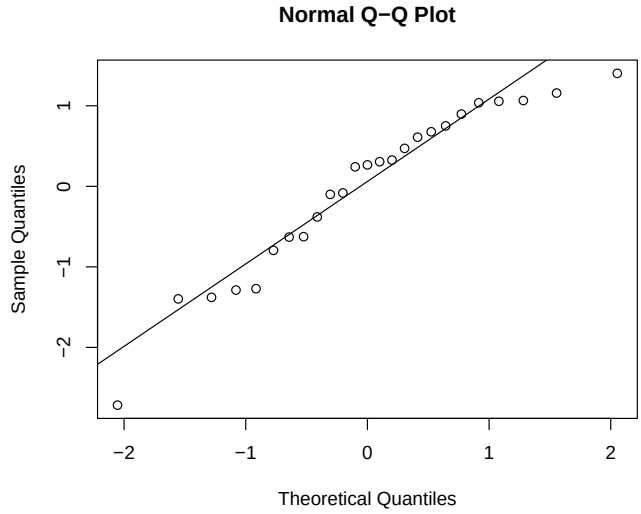


Figura 5.7: *qq-plot* dei residui standardizzati.

Come si può rimediare? I grafici precedenti suggeriscono che la variabile esplicativa può ancora spiegare una parte della variabilità del termine d'errore (della variabile risposta). Quindi il modello potrebbe essere migliorato. Una possibilità consiste nell'introdurre come ulteriore regressore la velocità del vento al quadrato. Si passerebbe quindi da una regressione semplice ad una regressione multipla. Una soluzione diversa può basarsi sulla ricerca di una trasformazione della variabile esplicativa, che migliori la linerità della relazione tra la variabile risposta e la variabile esplicativa. La ricerca di tale relazione può essere guidata dalla analisi del fenomeno che si sta studiando. Per esempio, si nota che la relazione tra velocità del vento e la quantità di corrente generata è monotona non decrescente. Tuttavia, si osserva che ad incrementi di velocità sopra una certa soglia, non sembrano corrispondere sostanziali guadagni in termini di energia elettrica. Quindi, potremmo considerare il reciproco del regressore, cioè $z_i = 1/x_i,\ i = 1, \ldots, n$, e considerare un modello del tipo

$$Y_i = \beta_1 + \beta_2 z_i + \varepsilon_i, \quad i = 1, \ldots, n.$$

In questo nuovo modello, ci si attende una stima negativa per β_2.

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

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: < 2.2e-16
```

La funzione `I()` nella formula permette di specificare la trasformazione da applicare alla variabile esplicativa. Nella fattispecie, si utilizza il reciproco.
I risultati precedenti mostrano significatività dei coefficienti e l'adattamento complessivo del modello, misurato in termini di R^2, indica un miglioramento rispetto al primo modello. Inoltre, i risultati delle analisi dei residui appaiono migliori, come mostrato dai grafici pertinenti riportati in Figura 5.8.

```
> par(mfrow = c(1, 2))
> vento.rstand.1 <- rstandard(vento.lm.1)
> vento.fit.1 <- fitted(vento.lm.1)
> plot(1/wind, vento.rstand.1)
> plot(vento.fit.1, vento.rstand.1)
```

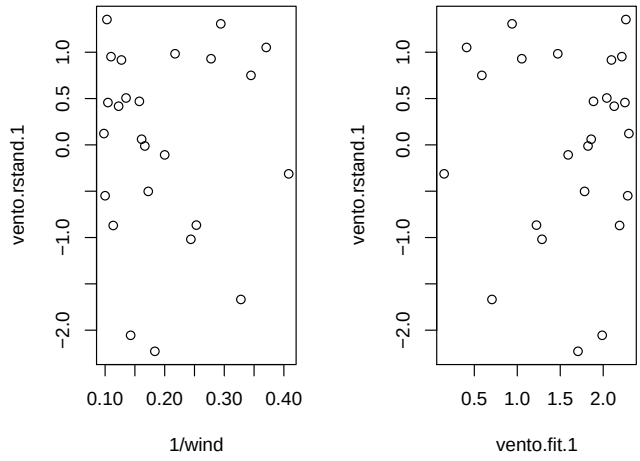


Figura 5.8: Grafico dei residui standardizzati rispetto ai `1/wind` e rispetto ai valori stimati dal modello.

Esercizio. Si completi l'analisi dei residui. ◇

È possibile ottenere previsioni della corrente elettrica prodotta dalla turbina anche per valori della velocità del vento non osservati, come, per esempio, valori esterni al campo di variazione osservato per la velocità. A tal fine, si può utilizzare la funzione `predict()`.

```
> new.wind <- 1:20
> vento.pred.1 <- predict(vento.lm.1, newdata = data.frame(wind = new.wind))
```

La funzione `predict()` è una generalizzazione della funzione `fitted()`. Quest'ultima fornisce i valori stimati del modello, $\hat{y}_i$, relativi ai valori x_i osservati; la funzione `predict()`, invece, permette di specificare, attraverso l'argomento `newdata`, valori non osservati per la variabile esplicativa, restituendo i corrispondenti valori previsti dal modello.
È possibile rappresentare, come in Figura 5.9 la curva stimata e i dati osservati con i comandi seguenti

```
> plot(new.wind, vento.pred.1, type = "l")
> points(wind, dc, col = 2)
```

o, in modo, alternativo, con i comandi

```
> plot(wind, dc, xlim = c(1, 20), ylim = c(0, 3))
> curve(predict(vento.lm.1, newdata = data.frame(wind = x)),
+     add = TRUE, col = 2)
```

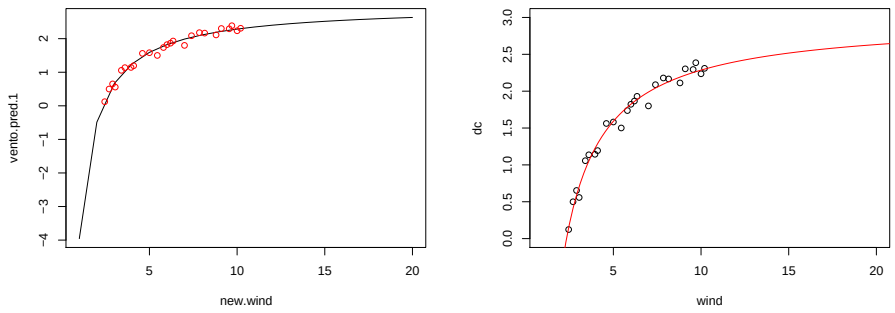


Figura 5.9: Retta stimata con `1/wind` come variabile esplicativa, trasformata nella scala originaria dei dati.

Capitolo 6

Test t di Student

6.1 Analisi dei dati FRUITFLY.DAT

I dati contenuti nel file `fruitfly.dat` si riferiscono alla fecondità di moschini della frutta *Drosophila melanogaster*, misurata come numero medio giornaliero di uova prodotte nei primi 14 giorni di vita. I dati sono relativi a 75 femmine, di cui 25 appartenenti alla linea genetica RS (resistente al DDT), 25 appartenenti alla linea genetica SS (suscettibile al DDT), 25 appartenenti ad una linea genetica non selezionata di controllo NS.

```
> Mosca <- read.table("fruitfly.dat", col.names = c("RS",
+     "SS", "NS"))
> attach(Mosca)
```

Si noti che i dati non sono nella forma di una matrice dei dati, dal momento che ciascuna riga del *data frame* `Mosca` contiene il valore della fecondità per 3 diversi moschini, ciascuno appartenente ad una delle 3 linee genetiche considerate.
Si vuole verificare in primo luogo se c'è differenza in media nella fecondità tra le prime due linee genetiche. Interessa poi anche valutare se c'è differenza in media tra la prima e la terza e tra la seconda e la terza linea. Tali ipotesi sono relative all'uguaglianza delle medie di due popolazioni; la loro verifica può essere effettuata utilizzando il test t di Student.
Per procedere, si deve verificare se le ipotesi di base, normalità e omoschedasticità delle popolazioni, sono soddisfatte. Un'idea sulla simmetria, dispersione, normalità della distribuzione dei dati nei due gruppi può essere ottenuta per via grafica (si vedano le Figure 6.1 e 6.2).

```
> boxplot(RS, SS)
```

```
> par(mfrow = c(2, 2))
> hist(RS)
> hist(SS)
> qqnorm(RS)
> qqline(RS)
> qqnorm(SS)
> qqline(SS)
```

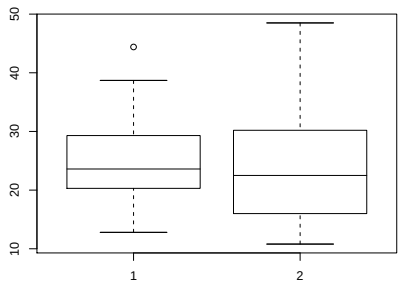


Figura 6.1: Diagramma a scatola per le linee RS e SS.

Sembrerebbe esserci una lieve asimmetria nella distribuzione di SS. Inoltre, anche la variabilità può sembrare a prima vista maggiore nel secondo gruppo. Per verificare l'ipotesi di omoschedasticità è possibile confrontare le varianze campionarie

```
> var(RS)

[1] 60.41007

> var(SS)

[1] 95.42293
```

Queste, in effetti, confermano l'informazione ottenibile dal confronto dei diagrammi a scatola. Assumendo che i dati per le linee RS e SS siano realizzazioni indipendenti, rispettivamente da una $N(\mu_{RS}, \sigma^2_{RS})$ e una $N(\mu_{SS}, \sigma^2_{SS})$, si può utilizzare il test del rapporto di verosimiglianza per verificare $H_0 : \sigma^2_{RS} = \sigma^2_{SS}$ contro $H_1 : \sigma^2_{RS} \neq \sigma^2_{SS}$. Il test si calcola utilizzando la funzione `var.test()`.

```
> var.test(RS, SS)

        F test to compare two variances

data:  RS and SS
F = 0.6331, num df = 24, denom df = 24, p-value =
0.2698
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.2789774 1.4366273
sample estimates:
ratio of variances
         0.633077
```

Il risultato indica che, contrariamente a quanto si poteva intuire dall'analisi grafica, le varianze delle due popolazioni possono essere considerate uguali.
La funzione che calcola il test t di Student a due campioni per saggiare l'ipotesi $H_0 : \mu_{RS} = \mu_{SS}$ è la funzione `t.test()`. L'assunzione di omoschedasticità deve essere esplicitamente dichiarata tramite l'argomento `var.equal`.

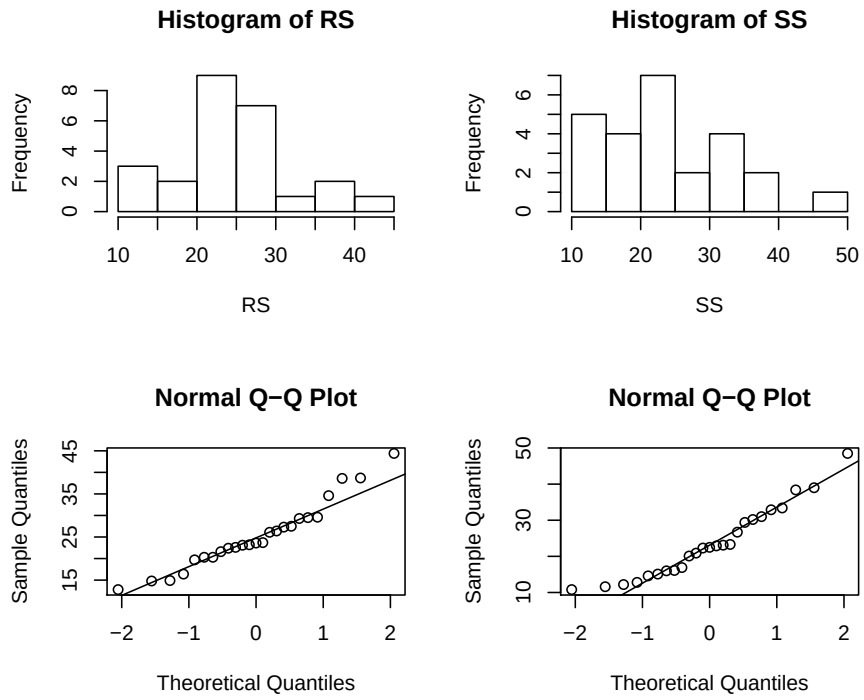


Figura 6.2: Istogrammi e *qq-plot* per le linee RS e SS.

```
> test <- t.test(RS, SS, var.equal = TRUE)
> test

        Two Sample t-test

data:  RS and SS
t = 0.6521, df = 48, p-value = 0.5175
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -3.391875  6.647875
sample estimates:
mean of x mean of y
   25.256    23.628
```

La quantità `t` riporta il valore osservato della statistica test, t^{oss}; `df` sono i gradi di libertà della distribuzione della statistica test sotto H_0 (in questo caso $n_1+n_2-2 = 25+25-2 = 48$); `p-value` è il livello di significatività osservato quando l'ipotesi alternativa è bilaterale $H_1 : \mu_{RS} \neq \mu_{SS}$, $\alpha^{\text{oss}} = 2\Pr_{H_0}(t > |t^{\text{oss}}|)$, con $t \sim t_{48}$ sotto H_0. Infatti:

```
> 2 * (1 - pt(test$statistic, test$parameter))

       t
0.517466
```

Poiché α^{oss} è piuttosto elevato, l'ipotesi nulla di uguaglianza delle medie μ_{RS} e μ_{SS} non è rifiutata. Si rifiuta H_0 contro H_1 al livello 0.05 se $|t^{\text{oss}}| > t_{48;0.975}$. Risulta

```
> qt(0.975, test$parameter)

[1] 2.010635
```

pertanto H_0 non viene rifiutata.
Si noti infine che nel caso di popolazioni eteroschedastiche, ovvero quando le varianze delle due popolazioni non possono essere assunte uguali, `R` permette di condurre un test approssimato per verificare l'ipotesi di uguaglianza delle medie delle due popolazioni. In questo caso, il comando è il seguente.

```
> t.test(campione1, campione2, var.equal = FALSE)
```

Come noto dalla teoria, il test t di Student può essere ricondotto ad un test di nullità di un coefficiente di regressione di un modello di regressione lineare semplice normale. Per fare ciò, è necessario scrivere i dati sotto forma di matrice dei dati 50×2, dove le colonne riportano per ciascuna unità statistica il valore della fecondità e la specie. È possibile creare il *data frame* attraverso i comandi seguenti.

```
> y <- c(RS, SS)
> group <- c(rep("RS", length(RS)), rep("SS", length(SS)))
> group <- factor(group)
> Mosca1 <- data.frame(y, group)
> rm(y, group)
> detach(Mosca)
> attach(Mosca1)
```

La funzione `factor()` esegue la codifica di variabili qualitative in variabili indicatrici. Nel nostro caso, associa alla variabile `group` una variabile indicatrice con due modalità numeriche in sostituzione delle modalità qualitative. Per vedere la codifica effettuata da `R` si può usare il seguente comando

```
> contrasts(group)

   SS
RS  0
SS  1
```

Questo significa la variabile indicatrice associata alla variabile `group` assume il valore 0 quando `group` assume il valore RS e il valore 1 quando `group` assume il valore SS. In sostanza, il fattore `group` è un indicatore di appartenenza al gruppo SS.

Quindi, indicata con x_i, $i = 1, \ldots, 50$, la variabile indicatrice sopra creata, che assume valore 1 se l'i-esima osservazione appartiene al gruppo SS e 0 altrimenti (ossia se appartiene al gruppo RS), e con y_i, $i = 1, \ldots, 50$, la fecondità dei moschini, è possibile costruire il modello di regressione lineare semplice normale

$$Y_i = \beta_1 + \beta_2 x_i + \varepsilon_i \,,$$

con ε_i variabili casuali normali $N(0, \sigma^2)$ indipendenti, $i = 1, \ldots, 50$. Il modello assume che per $i = 1, \ldots, 25$, ossia per i moschini RS, $Y_i \sim N(\beta_1, \sigma^2)$, mentre per $i = 26, \ldots, 50$, $Y_i \sim N(\beta_1 + \beta_2, \sigma^2)$. Si proceda quindi alla stima del modello.

```
> mosca.lm <- lm(y ~ group)
> summary(mosca.lm)

Call:
lm(formula = y ~ group)

Residuals:
    Min      1Q  Median      3Q     Max
-12.828  -6.435  -1.442   4.319  24.872

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   25.256      1.765  14.306   <2e-16 ***
groupSS       -1.628      2.497  -0.652    0.517
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

Residual standard error: 8.827 on 48 degrees of freedom
Multiple R-Squared: 0.00878,        Adjusted R-squared: -0.01187
F-statistic: 0.4252 on 1 and 48 DF,  p-value: 0.5175
```

L'ipotesi $H_0 : \mu_{RS} = \mu_{SS}$ è equivalente all'ipotesi $H_0 : \beta_2 = 0$. Si noti l'uguaglianza dei livelli di significatività osservati per i due test nei risultati delle due funzioni `t.test()` e `lm()`.

Esercizio. Si provi a ripetere l'analisi trasformando i dati mediante trasformazione logaritmica, specificando l'ipotesi posta sotto verifica. In particolare, si commenti l'effetto della trasformazione su simmetria e normalità. ◇

Esercizio. Si verifichi l'ipotesi di uguaglianza delle medie delle linee RS e NS. ◇

Esercizio. Alla luce dei risultati dell'analisi svolta all'inizio, si crei un unico gruppo con le prime due linee genetiche, RS e SS, e si confronti questo gruppo con la terza linea geneticha NS. ◇

6.2 Analisi dei dati CAPTO.DAT

I dati contenuti nel file `capto.dat` sono relativi a misurazioni della pressione sistolica e diastolica del sangue (in mm di mercurio) in un gruppo di 15 pazienti con lieve ipertensione, prima e dopo la somministrazione del farmaco *Captopril*. Si vuole verificare l'efficacia del farmaco nell'abbassare in media le due pressioni.

```
> Capto <- read.table("capto.dat")
> attach(Capto)
```

In questo caso non è possibile condurre un test t a due campioni per verificare, ad esempio, l'uguaglianza delle medie delle distribuzioni della pressione sistolica prima (Sp) e dopo (Sd) la somministrazione. Infatti, se anche l'ipotesi di normalità fosse adeguata, l'ipotesi di indipendenza delle osservazioni non può certamente esserlo, essendo Sp e Sd misurazioni della pressione sistolica effettuate in tempi diversi, ma sugli stessi soggetti. Siamo quindi in presenza di un unico campione di unità statistiche, osservato però in due momenti distinti. Tali dati vengono chiamati "dati appaiati".
Una possibile soluzione risiede nel considerare la variabile differenza $SD = Sd - Sp$. Infatti, assumendo che le coppie (Sd_i, Sp_i) siano realizzazioni indipendenti da una normale bivariata con componenti marginali $N(\mu_{Sd}, \sigma^2_{Sd})$ e $N(\mu_{Sp}, \sigma^2_{Sp})$, rispettivamente, e con coefficiente di correlazione ρ, allora l'insieme delle differenze $SD_i = Sd_i - Sp_i$, $i = 1, \ldots, 15$, è un campione casuale da una $N(\mu_{SD}, \sigma^2_{SD})$, con $\mu = \mu_{Sd} - \mu_{Sp}$ e $\sigma^2 = \sigma^2_{Sd} + \sigma^2_{Sp} - 2\rho\sigma^2_{Sd}\,\sigma^2_{Sp}$. Quindi, si può verificare l'ipotesi nulla $H_0 : \mu_{Sd} = \mu_{Sp}$, verificando l'ipotesi $H_0 : \mu_{SD} = 0$, attraverso un test t ad un campione aaplicato alle differenze SD_i, $i = 1, \ldots, 15$.

Si proceda quindi alla verifica grafica, mostrata in Figura 6.3, della normalità delle differenze SD_i, tenendo dovuto conto dell'esigua numerosità campionaria.

```
> SD <- Sd - Sp
> boxplot(SD)
> qqnorm(SD)
> qqline(SD)
```

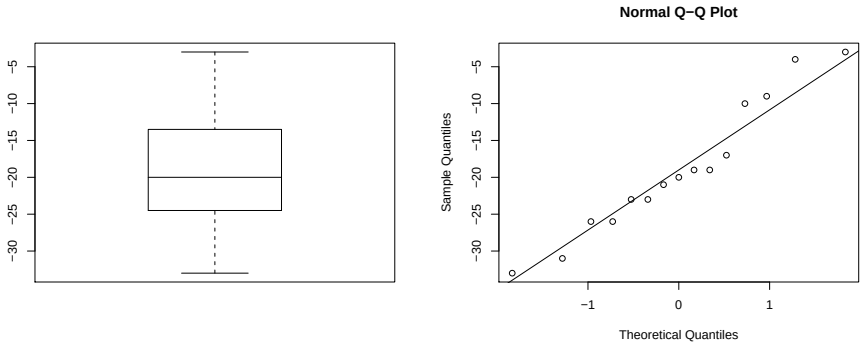


Figura 6.3: Istogramma e *qq-plot* per la variabile SD.

Si può procedere alla verifica dell'ipotesi $H_0 : \mu_{SD} = 0$.

```
> t.test(SD)

        One Sample t-test

data:  SD
t = -8.1228, df = 14, p-value = 1.146e-06
```

```
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 -23.93258 -13.93409
sample estimates:
mean of x
-18.93333
```

L'ipotesi nulla viene rifiutata. Il test appena condotto può essere sviluppato anche utilizzando la funzione `t.test()` sui dati di partenza. È necessario però specificare mediante l'argomento `paired` che si è in presenza di dati appaiati.

```
> t.test(Sd, Sp, paired = TRUE)

        Paired t-test

data:  Sd and Sp
t = -8.1228, df = 14, p-value = 1.146e-06
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -23.93258 -13.93409
sample estimates:
mean of the differences
              -18.93333
```

Il risultato è, ovviamente, identico a quello ottenuto dal test t ad un campione sulle differenze.
In realtà quello che interessa in questo caso è verificare se c'è stato un miglioramento, cioè se la pressione si è abbassata. È perciò più adeguato considerare un problema di verifica d'ipotesi con alternativa unilaterale:

$$H_0 : \mu_{SD} \geq 0$$
$$H_1 : \mu_{SD} < 0.$$

```
> t.test(Sd, Sp, paired = TRUE, alternative = "l")

        Paired t-test

data:  Sd and Sp
t = -8.1228, df = 14, p-value = 5.732e-07
alternative hypothesis: true difference in means is less than 0
95 percent confidence interval:
      -Inf -14.82793
sample estimates:
mean of the differences
              -18.93333
```

Essendo $\alpha^{\text{oss}} \doteq 0$, si rifiuta l'ipotesi nulla. Quindi, è ragionevole pensare che il farmaco determini, in media, un abbassamento della pressione. Per il livello di significatività $\alpha = 0.05$, la soglia della regione di rifiuto è data da

```
> qt(0.05, 14)

[1] -1.76131
```

Esercizio. Si ripeta l'analisi per la pressione diastolica. ◇

Capitolo 7

Modello di regressione lineare multipla

7.1 Analisi dei dati HOOK.DAT

I dati contenuti nel file `hook.dat`, raccolti nel XIX secolo dallo scienziato inglese Joseph Hooker sulle montagne dell'Himalaya, riportano le temperature di ebollizione dell'acqua (in gradi Farenheit) relative a diversi valori della pressione atmosferica (in pollici di mercurio). Si vuole studiare la relazione tra le due variabili.

```
> Hook <- read.table("hook.dat", col.names = c("temp",
+     "press"))
> attach(Hook)
```

I dati appaiono ordinati per temperature decrescenti. Si indichino con (y_i, x_i), $i = 1, \ldots, 31$, le 31 coppie di valori di temperatura e pressione. La temperatura y_i decresce al decrescere della pressione x_i. Questo suggerisce l'esistenza di un legame tra le due variabili.
Per esplorare graficamente la relazione si analizzi il diagramma di dispersione riportato in Figura 7.1.

```
> plot(temp ~ press)
```

Il diagramma di dispersione mostra una evidente relazione crescente tra le due variabili. Si ipotizza quindi un modello di regressione lineare semplice. In particolare, si assume che $y_1, \ldots, y_{31}$ siano realizzazioni di v.c. indipendenti Y_i con $Y_i = \beta_1 + \beta_2 x_i + \varepsilon_i$, con $\varepsilon_i \sim N(0, \sigma^2)$ indipendenti, $i = 1, \ldots, 31$. Il modello viene adattato con `R` tramite i seguenti comandi.

```
> hook.lm <- lm(temp ~ press)
> summary(hook.lm)

Call:
lm(formula = temp ~ press)

Residuals:
```

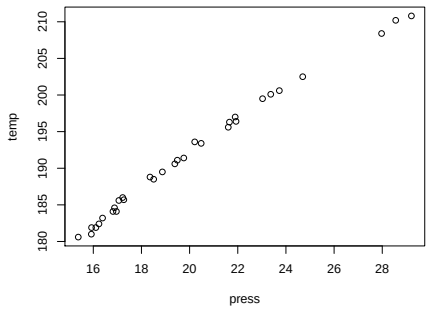


Figura 7.1: Diagramma di dispersione.

```
    Min      1Q  Median      3Q     Max
-1.6735 -0.6805  0.2203  0.5296  1.3976

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 146.67290    0.77641  188.91   <2e-16 ***
press         2.25260    0.03809   59.14   <2e-16 ***
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

Residual standard error: 0.806 on 29 degrees of freedom
Multiple R-Squared: 0.9918,        Adjusted R-squared: 0.9915
F-statistic:  3498 on 1 and 29 DF,  p-value: < 2.2e-16
```

Entrambi i coefficienti di regressione sono significativamente diversi da zero. Il coefficiente di determinazione (`Multiple R-Squared:  0.9918`) indica che il 99% della variabilità della temperatura è spiegata dalla sua relazione lineare con la pressione.
Utilizzando il comando

```
> abline(coef(hook.lm))
```

è possibile aggiungere al diagramma di dispersione la retta stimata, come mostrato in Figura 7.2.
Si passi ora all'analisi grafica dei residui, come in Figura 7.3.

```
> par(mfrow = c(2, 2))
> plot(hook.lm)
```

Il grafico dei residui rispetto ai valori predetti mostra un andamento parabolico. In particolare, per valori della variabile esplicativa agli estremi del suo campo di variazione, i valori della risposta sono sistematicamente sovrastimati, mentre per valori centrali la risposta è sottostimata.
Il *qq-plot* dei residui mostra qualche scostamento dalla normalità per i residui di segno positivo, suggerendo una possibile asimmetria nella distribuzione degi errori. Ciò può essere valutato con gli usuali strumenti grafici, come mostrato in Figura 7.4.

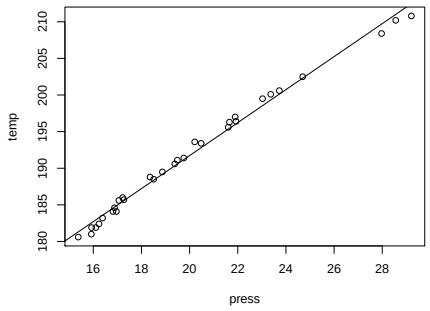


Figura 7.2: Diagramma di dispersione e retta stimata.

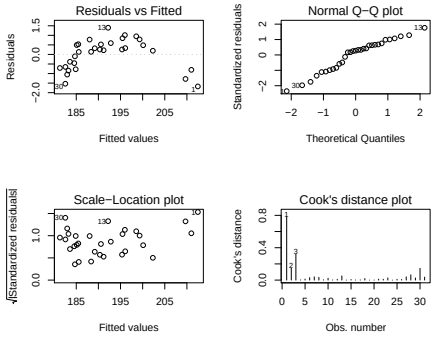


Figura 7.3: Analisi grafica dei residui.

```
> hook.rstand <- rstandard(hook.lm)
> hist(hook.rstand)
> boxplot(hook.rstand)
```

In conclusione, l'analisi dei residui non appare del tutto soddisfacente, nonostante il valore elevato di R^2 e la significatività dei coefficienti della regressione.
Per capire come individuare un modello con migliori caratteristiche di adattamento, può essere utile considerare il diagramma di dispersione dei residui rispetto alla variabile esplicativa, mostrato in Figura 7.5.

```
> plot(hook.rstand ~ press)
```

La Figura 7.5 mostra una relazione di tipo quadratico. Questo suggerisce che il modello potrebbe essere migliorato introducendo la pressione al quadrato come ulteriore variabile esplicativa. In questo modo si passa da un modello di regressione lineare semplice ad modello di regressione lineare multipla di tipo polinomiale. Il modello statistico diventa

$$Y_i = \beta_1 + \beta_2 x_i + \beta_3 x_i^2 + \varepsilon_i \,,$$

con $\varepsilon_i \sim N(0, \sigma^2)$, $i = 1, \ldots, n$. Il modello è adattato con `R` tramite i seguenti comandi.

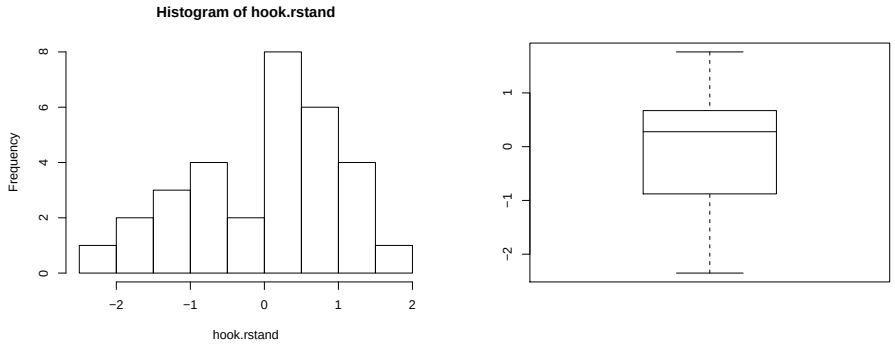


Figura 7.4: Istogramma e diagramma a scatola dei residui stadardizzati.

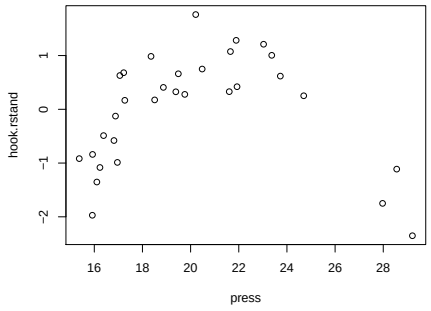


Figura 7.5: Diagramma di dispersione dei residui standardizzati rispetto alla variabile esplicativa.

```
> hook.lm.1 <- lm(temp ~ press + I(press^2))
> summary(hook.lm.1)

Call:
lm(formula = temp ~ press + I(press^2))

Residuals:
     Min       1Q   Median       3Q      Max
-0.79906 -0.26314 -0.01578  0.25139  0.73891

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 126.701623   2.112363  59.981  < 2e-16 ***
press         4.157627   0.199069  20.885  < 2e-16 ***
I(press^2)   -0.043754   0.004552  -9.612 2.29e-10 ***
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

Residual standard error: 0.3956 on 28 degrees of freedom
```

```
Multiple R-Squared: 0.9981,        Adjusted R-squared: 0.998
F-statistic:  7307 on 2 and 28 DF,  p-value: < 2.2e-16
```

La funzione `I( )` è necessaria perché la pressione al quadrato sia considerata come una variabile esplicativa.
L'oggetto `lm` contenuto in `hook.lm.1` si può anche ottenere a partire dall'oggetto `hook.lm` generato precedentemente con

```
> hook.lm.1 <- update(hook.lm, . ~ . + I(press^2))
```

La funzione `update()` permette di aggiornare il modello corrente (`hook.lm`), cambiandone alcune componenti. In questo caso, si è aggiornata la formula del modello, lasciando la temperatura come variabile risposta (si noti la presenza del punto a sinistra di `~`) e si è aggiunta l'esplicativa pressione al quadrato.
I coefficienti di regressione sono tutti significativamente diversi da zero.
Si passi al controllo dei residui del modello.

```
> par(mfrow = c(2, 2))
> plot(hook.lm.1)
```

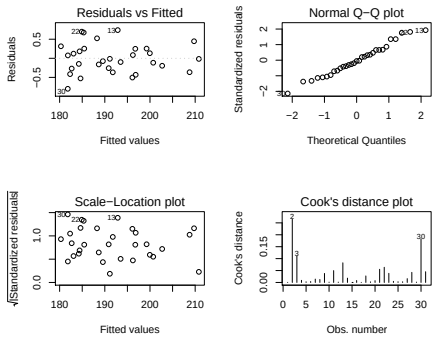


Figura 7.6: Analisi grafica dei residui.

Le analisi dei residui in Figura 7.6 appaiono notevolmente migliorate e complessivamente soddisfacenti.
Pertanto la relazione tra temperatura di ebollizione e pressione può essere ben descritta dal modello di regressione multipla polinomiale di secondo grado, come mostrato in Figura 7.7.

```
> plot(temp ~ press)
> hook.coeff.1 = coef(hook.lm.1)
> abline(hook.lm)
> curve(hook.coeff.1[1] + hook.coeff.1[2] * x +
+     hook.coeff.1[3] * x^2, add = TRUE, col = 2)
```

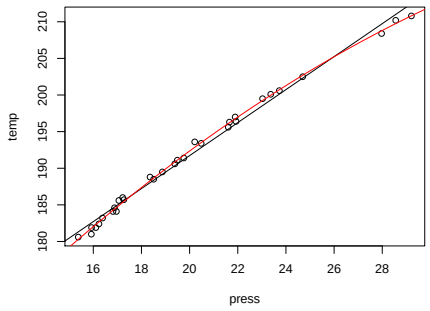


Figura 7.7: Confronto tra valori predetti per i modelli contenuti in `hook.lm` e `hook.lm.1`.

7.2 Analisi dei dati CHERRY.DAT

Si riconsiderino i dati contenuti nel file `cherry.dat`, riferiti a misurazioni rilevate su 31 alberi di ciliegio. Le variabili sono rispettivamente: diametro (a 4.5 piedi dal suolo, misurato in pollici), altezza (in piedi), volume (in piedi cubici) di legname utile. Si vuole studiare la dipendenza del volume di legno utile dall'altezza e dal diametro dell'albero.

```
> Ciliegi <- read.table("cherry.dat", col.names = c("diametro",
+     "altezza", "volume"))
> attach(Ciliegi)
```

È utile completare l'analisi grafica già parzialmente sviluppata nei laboratori precedenti.

```
> plot(volume ~ diametro)
> plot(volume ~ altezza)
> cor(diametro, volume)

[1] 0.9671194

> cor(altezza, volume)

[1] 0.5982497
```

La Figura 7.8 mostra una dipendenza del volume dal diametro; la relazione con l'altezza non è ben chiara.

```
> plot(diametro, altezza)
> cor(diametro, altezza)

[1] 0.5192801
```

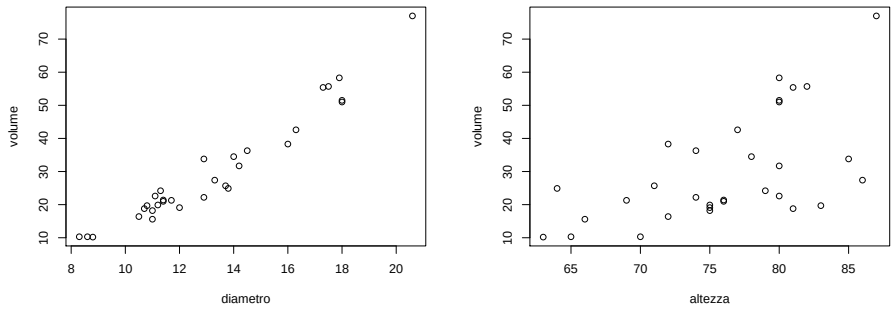


Figura 7.8: Diagrammi di dispersione tra la variabile risposta e le variabili esplicative.

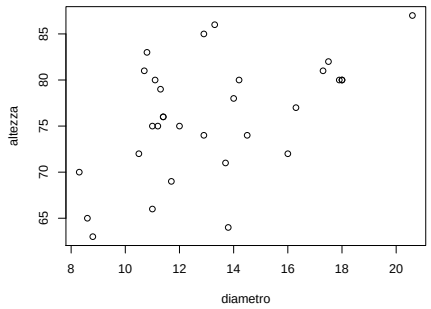


Figura 7.9: Diagramma di dispersione tra le variabili esplicative.

Anche fra diametro ed altezza c'è una leggera correlazione, come mostrato in Figura 7.9. Considerando il significato delle variabili che si stanno trattando e pensando che il tronco di un albero sia simile ad un cilindro (o ad un tronco di cono), è plausibile pensare che il suo volume possa essere rappresentabile mediante la relazione

$$\text{volume} \doteq k \cdot \text{diametro}^2 \cdot \text{altezza}^1 \,,$$

per una qualche costante k. Sembra opportuno quindi, per riportarsi ad una relazione additiva tra le variabili, trasformare i dati mediante la trasformata logaritmica, essendo

$$\log(\text{volume}) \doteq \log(k) + 2\log(\text{diametro}) + \log(\text{altezza}).$$

I diagrammi di dispersione tra le variabili su scala logaritmica sono riportati in Figura 7.10.

```
> plot(log(diametro), log(volume))
> plot(log(altezza), log(volume))
> cor(log(diametro), log(volume))

[1] 0.976665
```

```
> cor(log(altezza), log(volume))

[1] 0.6486377
```

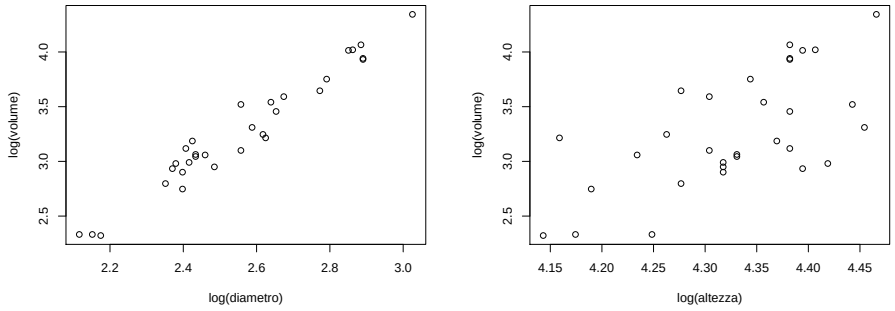


Figura 7.10: Diagrammi di dispersione su scala logaritmica.

Entrambi i coefficienti di correlazione sono aumentati. Si indichino con (y_i, x_{i2}, x_{i3}), $i = 1, \ldots, 31$, i valori dei logaritmi dei valori di volume, diametro e altezza, rispettivamente. Si desidera adattare un modello di regressione lineare multipla che assume i valori y_i realizzazioni di v.c. Y_i, con

$$Y_i \sim N(\beta_1 + \beta_2 x_{i2} + \beta_3 x_{i3}, \sigma^2), \qquad i = 1, \ldots, 31.$$

Il modello viene adattato ai dati tramite i comandi seguenti.

```
> ciliegi.lm <- lm(log(volume) ~ log(diametro) +
+     log(altezza))
> summary(ciliegi.lm)

Call:
lm(formula = log(volume) ~ log(diametro) + log(altezza))

Residuals:
      Min        1Q    Median        3Q       Max
-0.168561 -0.048488  0.002431  0.063637  0.129223

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   -6.63162    0.79979  -8.292 5.06e-09 ***
log(diametro)  1.98265    0.07501  26.432  < 2e-16 ***
log(altezza)   1.11712    0.20444   5.464 7.81e-06 ***
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

Residual standard error: 0.08139 on 28 degrees of freedom
Multiple R-Squared: 0.9777,        Adjusted R-squared: 0.9761
F-statistic: 613.2 on 2 and 28 DF,  p-value: < 2.2e-16
```

Tutti i coefficienti di regressione sono significativamente diversi da zero e R^2 è prossimo a uno. Si può passare all'analisi dei residui, risportata in Figura 7.11 e in Figura 7.12.

```
> ciliegi.rstand <- rstandard(ciliegi.lm)
> plot(log(diametro), ciliegi.rstand)
> plot(log(altezza), ciliegi.rstand)
```

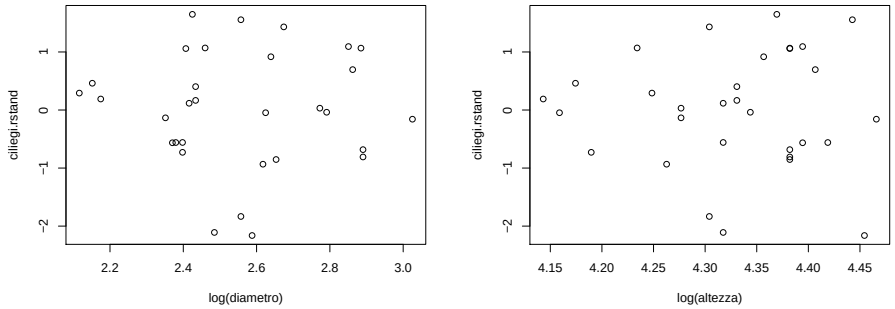


Figura 7.11: Diagrammi di dispersione tra i residui standardizzati e le variabili esplicative.

```
> par(mfrow = c(2, 2))
> plot(ciliegi.lm)
```

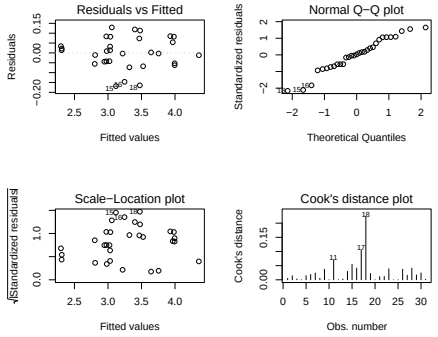


Figura 7.12: Analisi grafica dei residui.

I grafici in Figura 7.12 sembrano evidenziare la presenza di alcuni dati anomali, come ad esempio la 18-esima osservazione. Si può ripetere l'analisi eliminando tale osservazione.

```
> ciliegi.lm.1 <- lm(log(volume) ~ log(diametro) +
+     log(altezza), subset = -c(18))
> summary(ciliegi.lm.1)
```

```
Call:
lm(formula = log(volume) ~ log(diametro) + log(altezza), subset = -c(18))

Residuals:
     Min       1Q   Median       3Q      Max
-0.17500 -0.05706  0.00624  0.05940  0.11383

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   -7.18549    0.78061  -9.205 8.14e-10 ***
log(diametro)  1.95816    0.07051  27.770  < 2e-16 ***
log(altezza)   1.26100    0.19984   6.310 9.39e-07 ***
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

Residual standard error: 0.07565 on 27 degrees of freedom
Multiple R-Squared: 0.9814,        Adjusted R-squared:  0.98
F-statistic: 712.3 on 2 and 27 DF,  p-value: < 2.2e-16
```

L'argomento `subset` permette di eseguire l'analisi su uno specificato sottoinsieme di osservazioni. In questo caso si è indicato di prendere tutte le osservazioni, tranne la 18-esima.
Si passi all'analisi grafica dei residui (si veda la Figura 7.13).

```
> par(mfrow = c(2, 2))
> plot(ciliegi.lm.1)
```

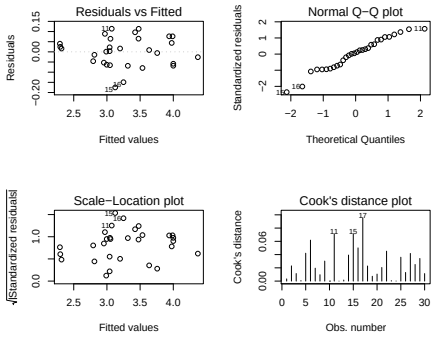


Figura 7.13: Analisi grafica dei residui.

Esercizio. Si ripeta l'analisi escludendo le altre osservazioni anomale. Si commentino i risultati ottenuti. ◇

Vista la natura del problema, ha senso chiedersi se nel modello in scala logaritmica i parametri relativi al logaritmo del diametro e al logaritmo dell'altezza possano essere pari rispettivamente a 2 e ad 1. Ciò implica, nella notazione introdotta in

precedenza e ponendo $\beta_1 = \log k$, verificare l'ipotesi $H_0 : \beta_2 = 2, \beta_3 = 1$. In altre parole, si tratta di confrontare il modello ridotto

$$y_i = \beta_1 + 2x_{i2} + 1x_{i3} + \varepsilon_i \,,$$

con il modello completo

$$y_i = \beta_1 + \beta_2 x_{i2} + \beta_3 x_{i3} + \varepsilon_i \,.$$

Per effettuare il confronto, si deve prima di tutto stimare il modello sotto l'ipotesi nulla.

```
> ciliegi.lm.0 <- lm(log(volume) - 2 * log(diametro) -
+     log(altezza) ~ 1)
```

Equivalentemente, si può utilizzare l'argomento `offset` della funzione `lm()`

```
> ciliegi.lm.0 <- lm(log(volume) ~ 1, offset = 2 *
+     log(diametro) + log(altezza))
```

Il test F che permette di verificare l'ipotesi nulla si basa sul confronto della devianza residua tra il modello stimato sotto H_0 (`ciliegi.lm.0`) e il modello completo (`ciliegi.lm`). Sotto l'ipotesi nulla, F ha distribuzione $F_{p-p_0,n-p}$, dove $p - p_0 = 3 - 1 = 0$ e $n - p = 31 - 1 = 30$. Tale test può essere condotto attraverso la funzione `anova()`, che permette il confronto tra modelli annidati

```
> anova(ciliegi.lm.0, ciliegi.lm)

Analysis of Variance Table

Model 1: log(volume) ~ 1
Model 2: log(volume) ~ log(diametro) + log(altezza)
  Res.Df      RSS Df Sum of Sq      F Pr(>F)
1     30 0.187686
2     28 0.185463  2  0.002222 0.1678 0.8464
```

Il livello di significatività osservato indica una forte evidenza in favore dell'ipotesi nulla, avvalorando la congettura sul valore dei coefficienti di regressione. Si ritornerà più in dettaglio sulla funzione `anova()` nei prossimi laboratori.

Capitolo 8

Costruzione del modello

8.1 Analisi dei dati HILLS.DAT

I dati contenuti nel file `hills.dat` riguardano il record (in minuti) registrato in 35 corse campestri effettuate sulle montagne scozzesi, la distanza coperta nelle corse (in miglia) e il dislivello affrontato (in piedi). Si desidera costruire un modello che metta in relazione il tempo record, trattato come variabile risposta, con la distanza ed il dislivello, trattati come variabili esplicative.

```
> Corse <- read.table("hills.dat")
> attach(Corse)
```

Si procede con un'analisi preliminare dei dati

```
> pairs(Corse)
```

Si ottengono i grafici riportati nella Figura 8.1. Si nota una relazione di tipo crescente tra la variabile risposta ed entrambe le variabili esplicative. La relazione appare particolarmente evidente se si considera la distanza.
Si possono calcolare anche i coefficienti di correlazione lineare semplice tra le variabili.

```
> cor(Corse)
```

```
           dist     climb      time
dist  1.0000000 0.6523461 0.9195892
climb 0.6523461 1.0000000 0.8052392
time  0.9195892 0.8052392 1.0000000
```

L'analisi dei coefficienti di correlazione conferma l'intuizione. Si noti che esiste anche una certa correlazione tra le due variabili esplicative.
Si procede quindi a costruire un modello di regressione lineare includendo prima la distanza come variabile esplicativa e successivamente, se dovesse risultare utile, il dislivello. Siano $y_1, \ldots, y_{35}$ i valori della risposta. Indicati con x_i i valori della distanza, si consideri in primo luogo il modello statistico che ipotizza che le y_i siano realizzazioni di v.c. Y_i tali che

$$Y_i = \beta_1 + \beta_2 x_i + \varepsilon_i \,,$$

$\varepsilon_i \sim N(0, \sigma^2)$ indipendenti, per $i = 1, \ldots, n$. Si adatti tale modello con `R`.

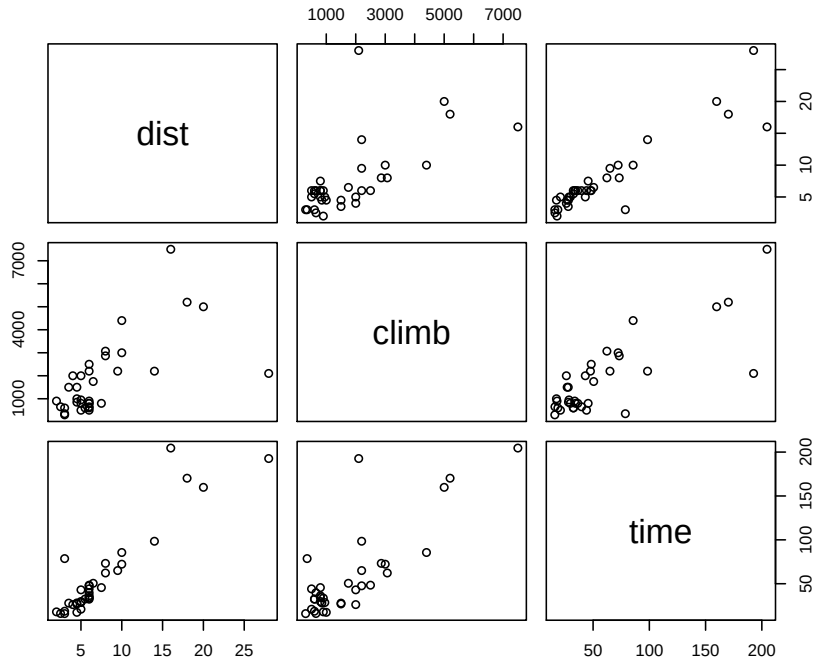


Figura 8.1: Matrice dei diagrammi di dispersione di `dist`, `climb` e `time`.

```
> corse.lm <- lm(time ~ dist)
> summary(corse.lm)

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 6.08e-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 DF,  p-value: 6.084e-15
```

Il coefficiente angolare β_2 risulta significativamente diverso da zero, mentre l'intercetta non risulta significativa. Ciò era atteso tenendo conto del significato delle variabili. Il coefficiente di determinazione R^2 indica che circa l'85% della variabilità

della risposta è spiegato attraverso la dipendenza lineare del tempo 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.
Per valutare se la variabile dislivello, non inclusa nel modello, possa contribuire a spiegare la variabilità residua, conviene analizzare il diagramma di dispersione dei residui standardizzati rispetto a tale variabile.

```
> corse.rstand <- rstandard(corse.lm)
> plot(climb, corse.rstand)
> identify(climb, corse.rstand, dimnames(Corse)[[1]])
```

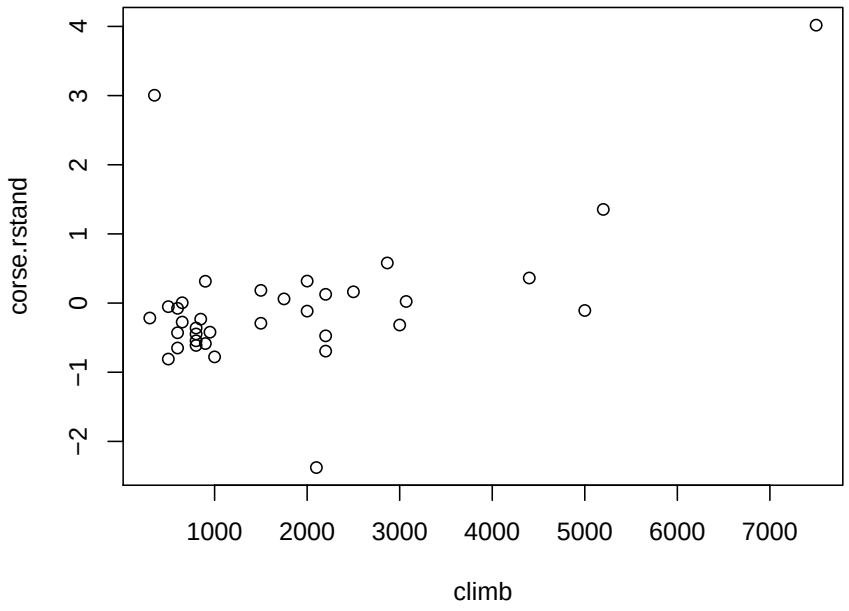


Figura 8.2: Diagramma di dispersione dei residui standardizzati rispetto a `climb`.

In effetti, il grafico in Figura 8.2 evidenzia una qualche relazione tra i residui e la variabile dislivello, a significare che il modello può probabilmente ancora essere migliorato tramite l'inserimento del dislivello.
Indicati con z_i i valori della variabile dislivello, $i = 1, \ldots, 35$, si consideri quindi il nuovo modello

$$Y_i = \beta_1 + \beta_2 x_i + \beta_3 z_i + \varepsilon_i \,, \qquad i = 1, \ldots, 35 \,,$$

Il modello può essere adattato aggiornando i risultati dell'analisi precedente.

```
> corse.lm.1 <- update(corse.lm, . ~ . + climb)
> summary(corse.lm.1)
```

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

Residuals:
    Min      1Q  Median      3Q     Max
-16.215  -7.129  -1.186   2.371  65.121

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -8.992039   4.302734  -2.090   0.0447 *
dist         6.217956   0.601148  10.343 9.86e-12 ***
climb        0.011048   0.002051   5.387 6.45e-06 ***
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

Residual standard error: 14.68 on 32 degrees of freedom
Multiple R-Squared: 0.9191,        Adjusted R-squared: 0.914
F-statistic: 181.7 on 2 and 32 DF,  p-value: < 2.2e-16
```

Anche il coefficiente relativo alla variabile dislivello risulta significativamente diverso da zero, permanendo la significatività di β_2. Si noti come, con l'inserimento di un'ulteriore variabile esplicativa, cambino sia la stima dell'intercetta sia quella del coefficiente di regressione relativo alla variabile distanza.
Il test sulla significatività di β_3 è equivalente al test per verificare l'ipotesi nulla che valga il modello ridotto, avente solo la distanza come variabile esplicativa, contro l'alternativa che valga il modello completo, avente come esplicative sia la distanza sia il dislivello. Trattandosi di modelli annidati, il test del rapporto di verosimiglianza è equivalente al test che rifiuta l'ipotesi nulla per valori grandi di

$$F = \frac{(\tilde{\sigma}^2 - \hat{\sigma}^2)/(p - p_0)}{\hat{\sigma}^2/(n - p)},$$

dove $\tilde{\sigma}^2$ e $\hat{\sigma}^2$ rappresentano le stime di massima verosimiglianza di σ^2 sotto il modello ridotto e sotto il modello completo, rispettivamente. Le quantità p, p_0 e n sono invece, nell'ordine, il numero di coefficienti di regressione nel modello completo, il numero di coefficienti di regressione nel modello ridotto e la numerosità campionaria. Qui $p = 3$, $p_0 = 2$ e $n = 35$. Sotto l'ipotesi nulla, la statistica F è realizzazione di una $F_{p-p_0,n-p}$.
In `R`, il test F è valutabile tramite la funzione `anova`.

```
> anova(corse.lm, corse.lm.1)

Analysis of Variance Table

Model 1: time ~ dist
Model 2: time ~ dist + climb
  Res.Df     RSS Df Sum of Sq      F    Pr(>F)
1     33 13141.6
```

```
2     32  6891.9  1    6249.7 29.019 6.445e-06 ***
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1
```

Risulta

$$F = \frac{(13141.6 - 6891.9)/(33 - 32)}{6891.9/32} = 29$$

con un livello di significatività osservato prossimo a zero. Il test suggerisce il rifiuto dell'ipotesi nulla e dunque sembra appropriato mantenere il modello completo.
Si può passare ora all'analisi dei residui.

```
> par(mfrow = c(2, 2))
> plot(corse.lm.1)
```

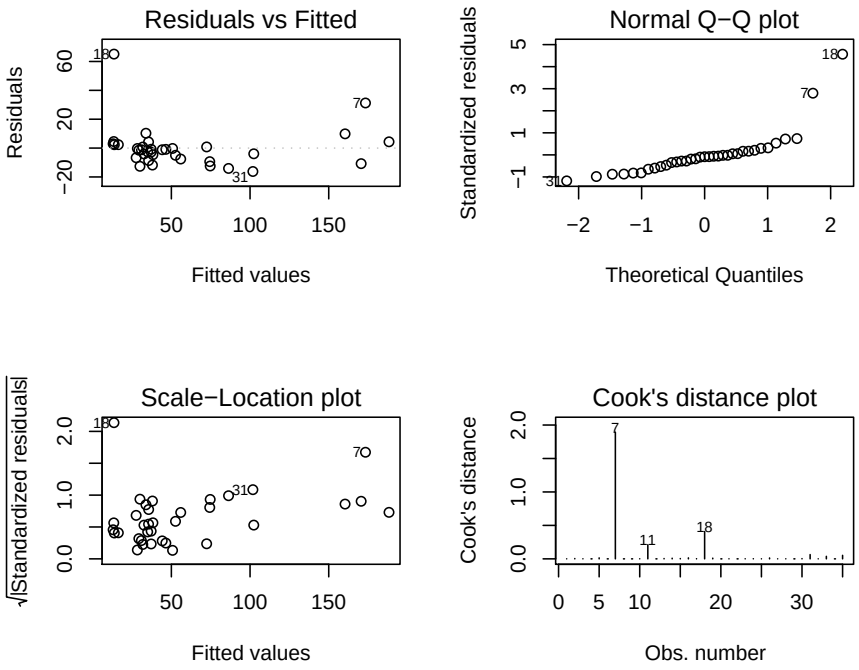


Figura 8.3: Analisi grafiche dei residui.

I grafici nella Figura 8.3 mostrano che alcune osservazioni paiono anomale. In particolare quelle di posizione 7 e 18.

```
> corse.rstand.1 <- rstandard(corse.lm.1)
> plot(fitted(corse.lm.1), corse.rstand.1)
> identify(fitted(corse.lm.1), corse.rstand.1, dimnames(Corse)[[1]])

NULL
```

```
NULL
```

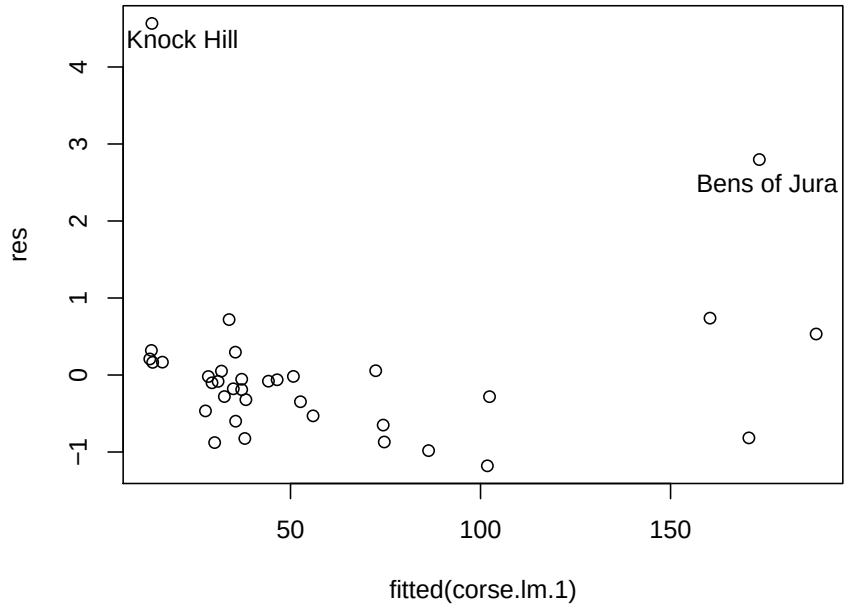


Figura 8.4: Identificazione grafica di osservazioni anomale.

Il grafico nella Figura 8.4 evidenzia ancora le osservazioni con residui particolarmente elevati. Per valutare l'effetto di queste osservazioni, si può ripetere l'analisi eliminandole, una alla volta.

Esercizio. Si ripeta l'adattamento del modello eliminando, nell'ordine, l'osservazione 18 e l'osservazione 7. Si commenti l'effetto sulle stime e sull'adattamento generale del modello dell'eliminazione delle due osservazioni. ◇

8.2 Analisi dei dati GASOLINE.DAT

Si desidera valutare l'effetto di diverse caratteristiche del petrolio grezzo sul quantitativo di benzina prodotta. A tal fine, è stato condotto un esperimento in cui, per 32 campioni di petrolio con diverse caratteristiche, è stata misurata la percentuale di benzina ottenuta dal processo di raffinazione.
Le variabili dell'insieme di dati contenuto nel *file* `gasoline.dat` sono nell'ordine, per $i = 1, \ldots, 32$,

- y_i, percentuale di benzina ottenuta dal petrolio grezzo;
- x_{i1}, gravità (in °API);
- x_{i2}, pressione del petrolio allo stato gassoso (in *psi*, *pounds/square inch*);

- x_{i3}, temperatura (in gradi Farenheit) alla quale il 10% di petrolio passa allo stato gassoso;
- x_{i4}, volatilità della benzina prodotta (misurato dal livello ASTM in gradi Farenheit).

Si vuole proporre un modello per valutare la dipendenza della percentuale y_i dalle altre variabili, opportunamente selezionate.

```
> Benz <- read.table("gasoline.dat", col.names = c("y",
+     "x1", "x2", "x3", "x4"))
> attach(Benz)
```

Si procede con un'analisi grafica preliminare dei dati. Un comando per vedere simultaneamente i grafici di tutte le possibili coppie di variabili è

```
> pairs(Benz)
```

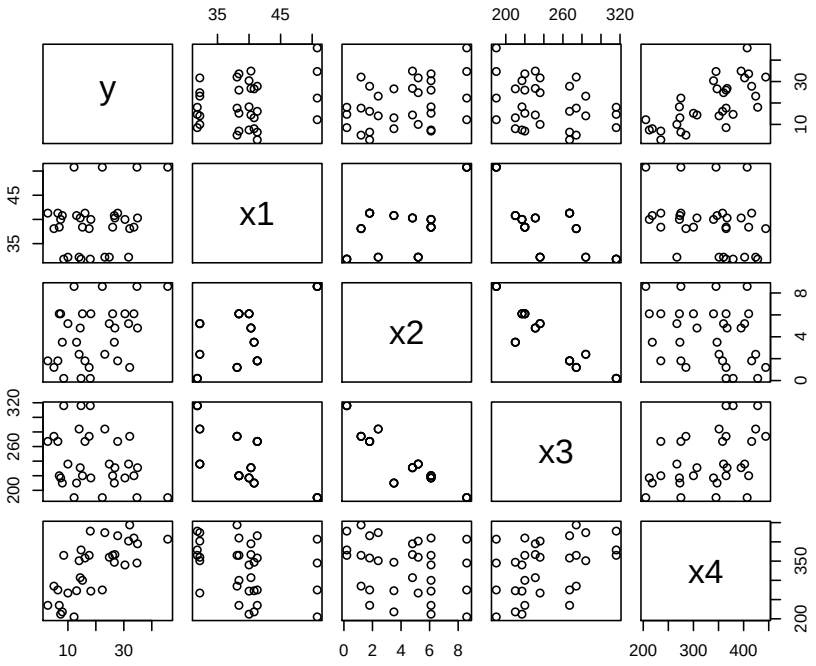


Figura 8.5: Matrice dei diagrammi di dispersione di `y` , `x1`, `x2`, `x3`, `x4`.

La prima riga di questa matrice di grafici si può ottenere nel modo seguente

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

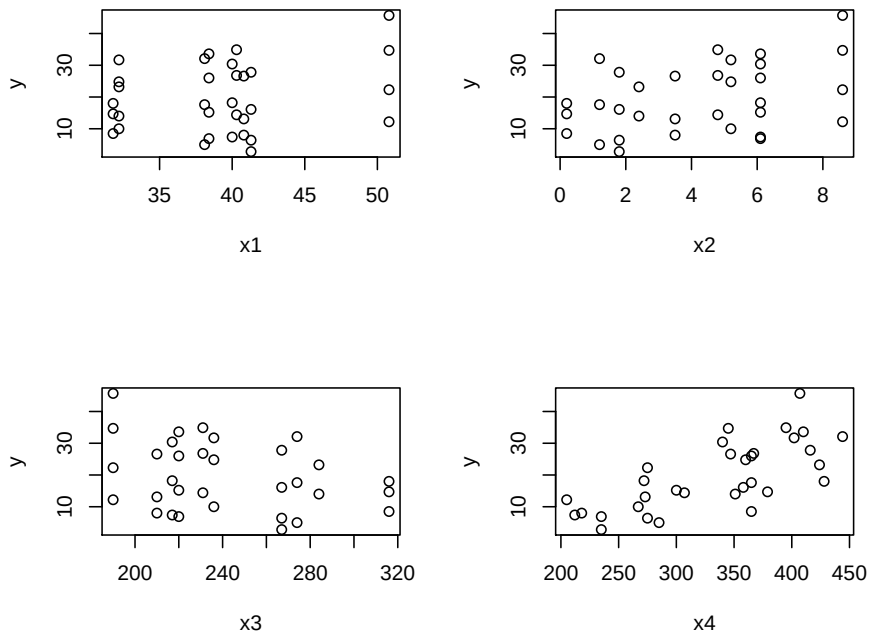


Figura 8.6: Diagrammi di dispersione di `y` rispetto a `x1`, `x2`, `x3`, `x4`.

I grafici sono indicativi solo della relazione fra la variabile risposta e ciascuna delle variabili esplicative, ma non danno informazione globale sulla dipendenza della risposta da tutte le variabili esplicative considerate simultaneamente.
È importante considerare anche la possibile dipendenza tra coppie di variabili esplicative.

```
> par(mfrow = c(1, 3))
> plot(x2, x3)
> plot(x1, x3)
> plot(x1, x2)
```

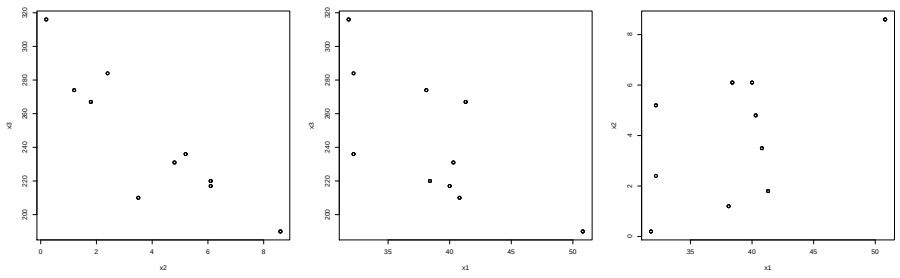


Figura 8.7: Diagrammi di dispersione tra coppie di variabili esplicative.

Si osserva una dipendenza lineare abbastanza marcata fra pressione e temperatura (x_2 e x_3) e fra gravità e temperatura (x_1 e x_3). Meno evidente è la relazione fra gravità e pressione (x_1 e x_2). È possibile quantificare attraverso la matrice di correlazione queste indicazioni grafiche.

```
> cor(Benz)
```

```
            y         x1         x2         x3         x4
y   1.0000000  0.2463260  0.3840706 -0.3150243  0.7115262
x1  0.2463260  1.0000000  0.6205867 -0.7001539 -0.3216782
x2  0.3840706  0.6205867  1.0000000 -0.9062248 -0.2979843
x3 -0.3150243 -0.7001539 -0.9062248  1.0000000  0.4122466
x4  0.7115262 -0.3216782 -0.2979843  0.4122466  1.0000000
```

Per quanto riguarda le correlazioni tra risposta ed esplicative, la più alta è quella relativa alla volatilità (x_4), come già ci si aspettava dai grafici precedenti. Le correlazioni tra le variabili esplicative confermano le indicazioni dei grafici precedenti.
Una marcata dipendenza lineare fra variabili esplicative, per esempio fra pressione e temperatura (x_2 e x_3), comporta problemi legati al fatto che nella matrice di regressione X le colonne corrispondenti sono "quasi linearmente dipendenti". In questi casi si parla di multicollinearità. In presenza di multicollineartà le stime sono numericamente instabili poiché la matrice $(X^\top X)$ è "quasi singolare". Per questo motivo, ci si aspetta che un buon modello non includa coppie di variabili esplicative fortemente correlate (come pressione e temperatura).
Si procede allora alla costruzione del modello. La prima variabile da includere è la volatilità (x_4) che ha il coefficiente di correlazione più elevato con la variabile risposta. In altre parole, si considerano $y_1, \ldots, y_{32}$ come realizzazioni di v.c. indipendenti Y_i, con $Y_i = \beta_0 + \beta_4 x_{i4} + \varepsilon_i$, dove $\varepsilon_i \sim N(0, \sigma^2)$, $i = 1, \ldots, 32$.

```
> benz.lm.4 <- lm(y ~ x4)
> summary(benz.lm.4)
```

```
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 DF,  p-value: 4.983e-06
```

Per la scelta della seconda variabile da introdurre, si utilizza la funzione `anova` che permette di confrontare due modelli annidati. Si confronterà il modello iniziale con ciascuno dei tre modelli ottenuti aggiungendo a `benz.lm.4` una delle prime tre variabili esplicative, `benz.lm.4`. Si sceglierà poi il modello a cui corrisponde il test F con livello di significatività osservato più piccolo, ossia che ha ridotto maggiormente la somma dei quadrati dei residui rispetto al modello ridotto $Y_i = \beta_0 + \beta_4 x_{i4} + \varepsilon_i$. Ovviamente, ciò andrà fatto purché tale livello di significatività sia più piccolo di una soglia prefissata, ad esempio 0.05.
Si adattino ora i tre modelli.

```
> benz.lm.41 <- update(benz.lm.4, . ~ . + x1)
> benz.lm.42 <- update(benz.lm.4, . ~ . + x2)
> benz.lm.43 <- update(benz.lm.4, . ~ . + x3)
```

Il confronto con il modello ridotto si ottiene con

```
> anova(benz.lm.4, benz.lm.41)

Analysis of Variance Table

Model 1: y ~ x4
Model 2: y ~ x4 + x1
  Res.Df     RSS Df Sum of Sq      F  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(benz.lm.4, benz.lm.42)

Analysis of Variance Table

Model 1: y ~ x4
Model 2: y ~ x4 + x2
  Res.Df     RSS Df Sum of Sq      F    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(benz.lm.4, benz.lm.43)

Analysis of Variance Table

Model 1: y ~ x4
Model 2: y ~ x4 + x3
  Res.Df     RSS Df Sum of Sq      F    Pr(>F)
1     30 1759.69
2     29  170.61  1   1589.08 270.11 3.111e-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 (`RSS`) più bassa, o equivalentemente la somma dei quadrati di regressione spiegata (`Sum Sq`) e il valore F (`F value`) più alti. Il modello migliore è quello che include le variabili volatilità e temperatura contenuto in `benz.lm.43`

$$Y_i = \beta_0 + \beta_3 x_{i3} + \beta_4 x_{i4} + \varepsilon_i,\ i = 1, \ldots, 32\,.$$

```
> summary(benz.lm.43)

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 3.11e-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 DF,  p-value: < 2.2e-16
```

La stessa selezione si poteva ottenere con il comando

```
> add1(benz.lm.4, . ~ . + x3 + x2 + x1)

Single term additions

Model:
y ~ x4
       Df Sum of Sq     RSS     AIC
<none>              1759.69  132.23
x3      1   1589.08  170.61   59.56
x2      1   1389.83  369.87   84.32
x1      1    897.75  861.95  111.39
```

Ogni riga rappresenta il modello in cui viene aggiunta al modello ridotto la relativa variabile. L'indice `AIC` rappresenta una trasformazione di `RSS` utile per confrontare la bontà di adattamento di modelli con un numero diverso di variabili esplicative. In particolare,

$$AIC = n \log \frac{RSS}{n} - n + 2p.$$

Un possibile criterio di selezione del modello si basa sulla minimizzazione di `AIC`.
Si ripeta ora lo stesso procedimento per valutare l'eventuale inclusione di una ulteriore variabile esplicativa tra gravità e pressione (x_1 e x_2).

```
> benz.lm.431 <- update(benz.lm.43, . ~ . + x1)
> benz.lm.432 <- update(benz.lm.43, . ~ . + x2)
> anova(benz.lm.43, benz.lm.431)

Analysis of Variance Table

Model 1: y ~ x4 + x3
Model 2: y ~ x4 + x3 + x1
  Res.Df    RSS Df Sum of Sq      F  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(benz.lm.43, benz.lm.432)

Analysis of Variance Table

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

I risultati portano a includere la gravità (x_1). In particolare, il secondo test F non risulta significativo ai livelli usuali poiché il livello di significatività osservato è pari a 0.1976. Ciò indica che non si ha un miglioramento significativo passando dal modello avente come esplicative volatilità e temperatura a quello che include anche la pressione (x_2). Questo era prevedibile, visto che il modello include già la temperatura che si è veduto essere fortemente correlata con la pressione. Il livello di significatività osservato relativo al primo confronto (0.03844) è comunque indice di un test significativo al livello 0.05.
Le principali quantità relative all'adattamento del modello

$$Y_i = \beta_0 + \beta_1 x_{i1} + \beta_3 x_{i3} + \beta_4 x_{i4} + \varepsilon_i\,, \qquad i = 1, \ldots, 32\,,$$

si ottengono con il comando `summary`:

```
> summary(benz.lm.431)

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 DF,  p-value: < 2.2e-16
```

Si noti che ora l'intercetta non è più significativa ($\alpha^{\text{oss}} = 0.5811$) e può dunque essere eliminata. Il modello risultante è

$$Y_i = \beta_1 x_{i1} + \beta_3 x_{i3} + \beta_4 x_{i4} + \varepsilon_i\,, \qquad i = 1, \ldots, 32\,,$$

e viene adattato tramite

```
> benz.lm.431bis <- update(benz.lm.431, . ~ . -
+     1)
> summary(benz.lm.431bis)

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 DF,  p-value: < 2.2e-16
```

Si noti che il coefficiente di determinazione R^2, nel caso di modello senza intercetta, è calcolato con la formula seguente

$$R^2 = \frac{\sum_{i=1}^n \hat{y}_i^2}{\sum_{i=1}^n y_i^2}\,,$$

che non è confrontabile con l'usuale coefficiente R^2 calcolato in un modello con intercetta. Tuttavia, continua a valere la relazione

$$F = \frac{R^2/p}{(1-R^2)/(n-p)}$$

che, sotto l'ipotesi nulla, ha distribuzione $F_{p,n-p}$.
Quest'ultimo potrebbe essere un modello adeguato. Ma si può anche decidere di non considerare la variabile x_1 significativa e proporre invece il modello

$$Y_i = \beta_0 + \beta_3 x_{i3} + \beta_4 x_{i4} + \varepsilon_i\,, \qquad i = 1, \ldots, 32\,,$$

i cui risultati di adattamento sono contenuti in `benz.lm.43`.
Un procedimento alternativo per la selezione delle variabili da includere nel modello è il seguente. Si considera inizialmente il modello di regressione lineare multipla che include tutte le variabili esplicative

$$Y_i = \beta_0 + \beta_2 x_{i2} + \beta_3 x_{i3} + \beta_4 x_{i4} + \varepsilon_i\,, \qquad i = 1, \ldots, 32\,.$$

L'adattamento avviene con

```
> benz.lm.1234 <- lm(y ~ x1 + x2 + x3 + x4)
> summary(benz.lm.1234)

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 DF,  p-value: < 2.2e-16
```

Dai livelli di significatività osservati relativi ai test sui singoli coefficienti di regressione, si nota che alcuni di essi non sono significativi. Le variabili corrispondenti si eliminano una alla volta, iniziando da quella cui corrisponde il più grande livello di significatività osservato (purché questo sia più grande di una soglia minima prefissata, ad esempio 0.05). In questo caso si elimina l'intercetta.

```
> benz.lm.1234bis <- update(benz.lm.1234, . ~ . -
+     1)
> summary(benz.lm.1234bis)

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 4.45e-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 DF,  p-value: < 2.2e-16
```

Il coefficiente relativo alla pressione (x_2) è ancora non significativo. Escludendo anche tale variabile si ottiene esattamente il modello selezionato con l'analisi precedente, i cui risultati di adattamento sono contenuti in `benz.lm.431bis`.
Analogo al comando `add1`, esiste anche il comando `drop1` che toglie dal modello una variabile esplicativa alla volta. In particolare, si elimina la variabile che comporta l'aumento più piccolo (non significativo) della somma dei quadrati dei residui.
Si può ora passare all'analisi dei residui per il modello

$$Y_i = \beta_1 x_{i1} + \beta_3 x_{i3} + \beta_4 x_{i4} + \varepsilon_i\,, \qquad i = 1, \ldots, 32\,.$$

```
> benz.rstand.431bis <- rstandard(benz.lm.431bis)
> par(mfrow = c(1, 3))
> hist(benz.rstand.431bis)
> boxplot(benz.rstand.431bis)
> qqnorm(benz.rstand.431bis)
> qqline(benz.rstand.431bis)
```

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

```
> par(mfrow = c(1, 3))
> plot(x1, benz.rstand.431bis)
> plot(x3, benz.rstand.431bis)
> plot(x4, benz.rstand.431bis)
```

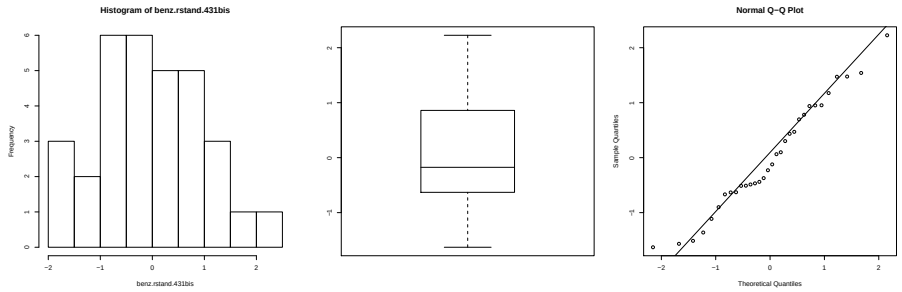


Figura 8.8: Analisi dei residui del modello `benz.lm.431bis`

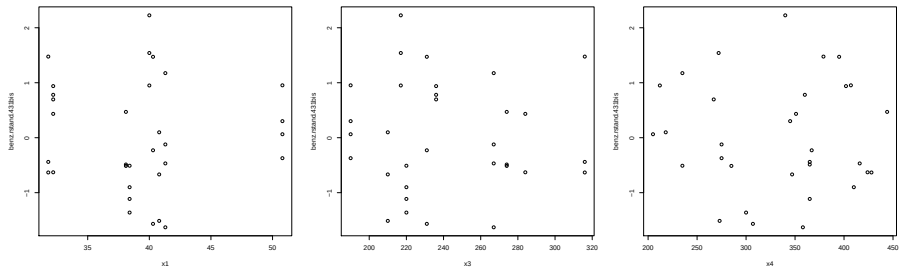


Figura 8.9: Diagrammi di dispersione dei residui di `benz.lm.431bis` rispetto alle variabili esplicative incluse nel modello.

Per assicurarsi di aver eliminato effettivamente una variabile non importante

```
> plot(x2, benz.rstand.431bis)
```

Esercizio. Si analizzino i residui relativi all'adattamento del modello

$$Y_i = \beta_0 + \beta_3 x_{i3} + \beta_4 x_{i4} + \varepsilon_i$$

, $i = 1, \ldots, 32$, $\varepsilon_i \sim N(0, \sigma^2)$, indipendenti. $\diamond$

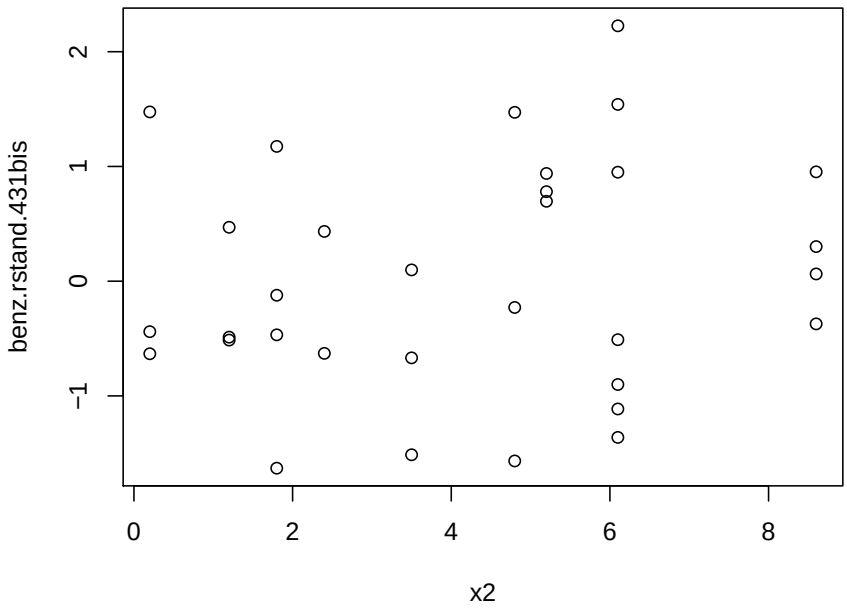


Figura 8.10: Diagrammi di dispersione dei residui di `benz.lm.431bis` rispetto alla variabile `x2` non inclusa nel modello.

Capitolo 9

Analisi della varianza ad un fattore

9.1 Analisi dei dati STURDY.DAT

I dati riportati nel file `sturdy.dat` si riferiscono ad un esperimento effettuato per studiare la resistenza allo strappo di diverse marche di impermeabili. Capi di cinque diverse marche sono stati sottoposti alla stessa sollecitazione. La resistenza allo strappo è stata misurata con il tempo (in minuti e frazioni decimali di minuti) intercorso tra la sollecitazione e lo strappo. I dati ottenuti sono

Marca A:	2.34	2.46	2.83	2.04	2.69	
Marca B:	2.64	3.00	3.19	3.83		
Marca C:	2.61	2.07	2.80	2.58	2.98	2.30
Marca D:	1.32	1.62	1.92	0.88	1.50	1.30
Marca E:	0.41	0.83	0.58	0.32	1.62	

Si desidera rispondere ai seguenti quesiti.

1. Esistono differenze di resistenza tra le varie marche?
2. È possibile dire se le marche A, B, C sono più resistenti, in media, delle marche D, E?

In primo luogo si acquisiscono i dati

```
> Imp <- read.table("sturdy.dat")
> names(Imp) <- "tempo"
```

Il *file* contiene tutti i dati in sequenza. Andrà aggiunta l'informazione che i primi 5 valori si riferiscono alla marca A, i secondi 4 alla B, eccetera. Si costruisce a tal fine la variabile qualitativa `marca` contenente la marca dell'impermeabile che successivamente viene trasformata in fattore e unita al *dataframe*.

```
> marca <- c(rep("A", 5), rep("B", 4), rep("C",
+     6), rep("D", 6), rep("E", 5))
> marca <- factor(marca)
> marca
```

```
 [1] A A A A A B B B B C C C C C C D D D D D D E E E E E
Levels: A B C D E
```

```
> Imp <- data.frame(Imp, marca)
> attach(Imp)
```

Si ricorda che il comando `factor` esegue la codifica di variabili qualitative in variabili indicatrici (*dummy*) delle diverse modalità, omettendo una modalità per l'identificabilità. In genere è la prima modalità ad essere omessa. Si può controllare la codifica utilizzata con

```
> contrasts(marca)
```

```
  B C D E
A 0 0 0 0
B 1 0 0 0
C 0 1 0 0
D 0 0 1 0
E 0 0 0 1
```

Ciò significa che sono state create 4 variabili indicatrici, indicate da `R` con `B`, `C`, `D` e `E`, che corrispondono, nell'ordine, a
$x_{iB} = 1$ se la marca dell'osservazione i-esima è la B e zero altrimenti,
$x_{iC} = 1$ se la marca dell'osservazione i-esima è la C e zero altrimenti,
$x_{iD} = 1$ se la marca dell'osservazione i-esima è la D e zero altrimenti,
$x_{iE} = 1$ se la marca dell'osservazione i-esima è la E e zero altrimenti,
per $i = 1, \ldots, 26$.
Pur tenendo conto dell'esiguo numero di osservazioni per gruppo, è utile come primo passo un'analisi di tipo grafico.

```
> plot(tempo ~ marca)
```

Il grafico riportato nella Figura 9.1 mostra una evidente differenza tra le marche in termini di mediane. In particolare, la marca B ha la mediana più elevata. Anche se a livello del tutto indicativo, la variabilità entro i gruppi parrebbe comparabile. La numerosità così esigua dei gruppi non permette di verificare la normalità, ad esempio tramite diagrammi quantile-quantile. Quindi, non sembra che ci siano elementi per mettere in discussione l'assunzione che le osservazioni siano realizzazioni di variabili casuali normali indipendenti con medie e varianze non necessariamente uguali nei 5 gruppi. In altri termini, indicata con y_{ij} l'osservazione i-esima del gruppo j-esimo, $j = 1, \ldots, 5$, si assume che le y_{ij} siano realizzazioni di v.c. indipendenti $N(\mu_j, \sigma_j^2)$.
Sulla base di tali assunzioni, si può costruire un test del rapporto di verosimiglianza per verificare l'ipotesi nulla di omoschedasticità, $H_0 : \sigma_1^2 = \cdots = \sigma_5^2$, ossia di uguaglianza delle varianze. Tale test (o meglio una sua funzione monotona) è noto come test di Bartlett di omogeneità delle varianze. In `R` è effettuato con il comando `bartlett.test`. La sintassi può essere indifferentemente in forma di formula (`bartlett.test(tempo~marca)`) oppure mettendo il nome della variabile risposta e il nome del fattore, separati da virgola

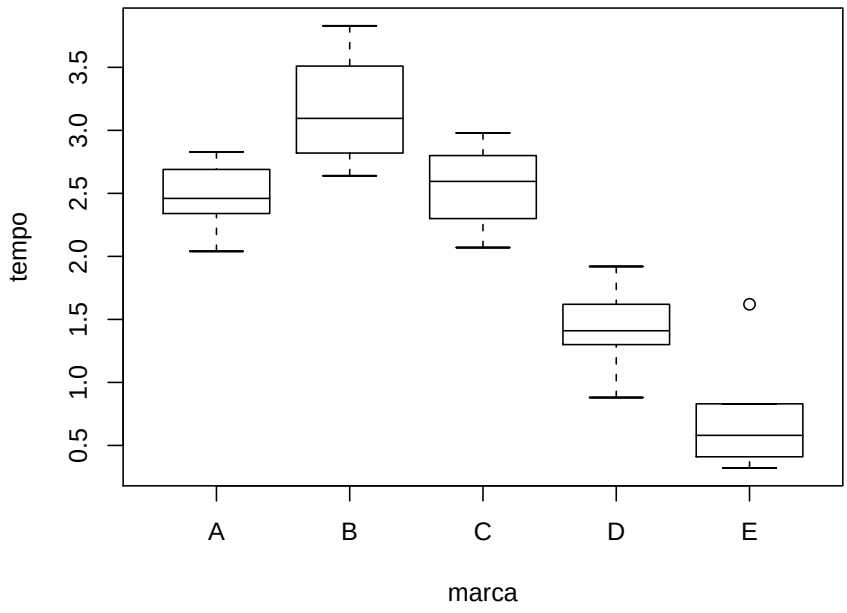


Figura 9.1: Diagrammi a scatola di `tempo` per le 5 marche.

```
> bartlett.test(tempo, marca)

        Bartlett test for homogeneity of variances

data:  tempo and marca
Bartlett's K-squared = 1.8016, df = 4, p-value =
0.7722
```

Il test porta all'accettazione dell'ipotesi nulla, ossia le varianze dei gruppi si possono considerare uguali.
L'osservazione circa la diversità delle mediane, unita all'ipotesi di normalità della variabile risposta, porta a ipotizzare che esista una differenza in termini di medie.
Si assuma che i valori y_i della variabile risposta (tempo) siano realizzazioni di v.c. Y_i con

$$Y_i = \beta_1 + \beta_2 x_{iB} + \beta_3 x_{iC} + \beta_4 x_{iD} + \beta_5 x_{iE} + \varepsilon_i \,,$$

con $\varepsilon_i \sim N(0, \sigma^2)$ indipendenti, $i = 1, \ldots, 26$. L'ipotesi di uguaglianza delle medie è allora formulabile come

$$H_0 \,:\, \beta_2 = \beta_3 = \beta_4 = \beta_5 = 0 \,.$$

Il test F corrispondente è

$$F = \frac{(\tilde{\sigma}^2 - \hat{\sigma}^2)/(5-1)}{\hat{\sigma}^2/(26-5)} = \frac{R^2/4}{(1-R^2)/21} \,,$$

dove $\hat{\sigma}^2$ rappresenta la stima di massima verosimiglianza di σ^2 sotto il modello completo e $n\tilde{\sigma}^2 = \sum_{i=1}^{n}(y_i - \bar{y})^2$. Sotto l'ipotesi nulla, la statistica F è realizzazione di una $F_{4,21}$.
Il modello precedente viene adattato con `R` utilizzando la funzione `lm`:

```
> imp.lm <- lm(tempo ~ marca)

> summary(imp.lm)

Call:
lm(formula = tempo ~ marca)

Residuals:
      Min        1Q    Median        3Q       Max
-0.543333 -0.235500  0.005667  0.212667  0.868000

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.47200    0.17929  13.788 5.40e-12 ***
marcaB       0.69300    0.26893   2.577 0.017583 *
marcaC       0.08467    0.24276   0.349 0.730733
marcaD      -1.04867    0.24276  -4.320 0.000302 ***
marcaE      -1.72000    0.25355  -6.784 1.04e-06 ***
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

Residual standard error: 0.4009 on 21 degrees of freedom
Multiple R-Squared: 0.8433,        Adjusted R-squared: 0.8135
F-statistic: 28.26 on 4 and 21 DF,  p-value: 3.475e-08
```

Il risultato indica che l'ipotesi nulla viene rifiutata. Infatti la statistica F è pari a 28.26 con un livello di significatività osservato prossimo a zero.
Per $j = 2, \ldots, 5$, il parametro β_j rappresentano la differenza tra la media del gruppo j (marca B, C, D o E) e quella del primo gruppo (marca A). La stima dell'intercetta, $\hat{\beta}_1$ è pari alla media del campione di osservazioni relative alla marca A. Per $j = 2, \ldots, 5$ la stima $\hat{\beta}_j$ relativa alla j-sima componente di $\hat{\beta} = (X^\top X)^{-1} X^\top y$ risulta pari alla differenza tra la media aritmetica delle osservazioni relative alla marca B, C, D, E, rispettivamente, e la media del campione di osservazioni relative alla marca A. Ad esempio, $\hat{\beta}_2$ è pari a 0.693. Ciò vuol dire che la stima della media nel gruppo B è pari a 2.472+0.693= 3.165.
I test sui singoli coefficienti equivalgono a test di eguaglianza delle medie di ciascuna marca con la media relativa alla marca A. Risulta che β_3 non è significativamente diverso da zero, mentre $H_0 : \beta_2 = 0$ viene accettata al livello 0.01, ma non al livello 0.05. Quindi, si potrebbe concludere che le medie relative alle marche A, B e C non sono significativamente diverse tra loro.
Se si è interessati solamente all'ipotesi complessiva $H_0 : \beta_2 = \beta_3 = \beta_4 = \beta_5 = 0$, si può effettuare direttamente l'analisi della varianza, con il comando

```
> imp.aov <- aov(tempo ~ marca)
> imp.aov

Call:
   aov(formula = tempo ~ marca)

Terms:
                    marca Residuals
Sum of Squares  18.168119  3.375127
Deg. of Freedom         4        21

Residual standard error: 0.4008994
Estimated effects may be unbalanced
```

Per avere informazioni circa l'esito del test F si utilizza la funzione `summary`

```
> summary(imp.aov)

            Df  Sum Sq Mean Sq F value    Pr(>F)
marca        4 18.1681  4.5420  28.261 3.475e-08 ***
Residuals   21  3.3751  0.1607
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1
```

La funzione presenta la classica tabella di scomposizione della varianza. La quantità indicata con `Pr(>F)` indica il livello di significatività osservato del test e coincide con il livello di significatività osservato del test F che si trova in `summary(imp.lm)`. Un test con livello fissato 0.05 rifiuta H_0 per valori di F maggiori del quantile di livello 0.95 di una distribuzione $F_{4,21}$. Questo valore è

```
> qf(0.95, 4, 21)
[1] 2.8401
```

I risultati indicano una forte evidenza contro l'ipotesi nulla di uguaglianza delle medie dei gruppi.

Per rispondere al secondo quesito, si possono accorpare i gruppi A, B, C ed i gruppi D, E per verificare poi la significatività della differenza tra le medie delle due nuove popolazioni, ossia quella relativa alle marche A, B e C e quella relativa alle marche D ed E.
Per fare questo si crea una variabile dicotomica `marca.1` le cui modalità corrispondono ai due nuovi gruppi e si aggiunge al *dataframe*.

```
> detach()
> marca.1 = NULL
> marca.1[(marca == "A") | (marca == "B") | (marca ==
+     "C")] = "G1"
> marca.1[!((marca == "A") | (marca == "B") | (marca ==
```

```
+     "C"))] = "G2"
> marca.1 = factor(marca.1)
> Imp <- data.frame(Imp, marca.1)
> attach(Imp)
```

Per valutare la significatività della differenza tra le due medie, si può utilizzare il test t di Student. Non è possibile in questo caso applicare ancora l'analisi della varianza invece del test t perché l'ipotesi alternativa è chiaramente unilaterale.

Esercizio. Prima di effettuare il test si verifichino le assunzioni che sono alla base del test t. ◇

Quindi

```
> t.test(tempo[marca.1 == "G1"], tempo[marca.1 ==
+     "G2"], alternative = "greater", var.equal = TRUE)

        Two Sample t-test

data:  tempo[marca.1 == "G1"] and tempo[marca.1 == "G2"]
t = 8.0229, df = 24, p-value = 1.500e-08
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 1.237152      Inf
sample estimates:
mean of x mean of y
 2.690667  1.118182
```

L'ipotesi nulla è decisamente rifiutata a favore dell'alternativa, che prevede che la resistenza media del gruppo di marche G1 sia maggiore della resistenza media del gruppo G2.

9.2 Analisi dei dati RATS.DAT

I dati contenuti nel file `rats.dat`, si riferiscono ad un disegno fattoriale 3×4 completamente randomizzato sull'effetto di agenti tossici ed antidoti. Si considerano 3 tipi di veleno (I, II, III) e 4 antidoti (A, B, C, D). Ogni combinazione veleno-antidoto viene sperimentata su 4 cavie. Per ogni cavia, viene osservato il tempo di sopravvivenza espresso in decine di ore. I dati, pertanto, risultano così costruiti.

Antidoto	Veleno				
A	I	0.31	0.45	0.46	0.43
	II	0.36	0.29	0.40	0.23
	III	0.22	0.21	0.18	0.23
B	I	0.82	1.10	0.88	0.72
	II	0.92	0.61	0.49	1.24
	III	0.30	0.37	0.38	0.29
C	I	0.43	0.45	0.63	0.76
	II	0.44	0.35	0.31	0.40
	III	0.23	0.25	0.24	0.22
D	I	0.45	0.71	0.66	0.62
	II	0.56	1.02	0.71	0.38
	III	0.30	0.36	0.31	0.33

```
> Topi <- read.table("rats.dat", header = TRUE)
```

Si noti che i dati sono già organizzati sotto forma di matrice dei dati. La variabile `veleno` rappresenta il veleno somministrato e la variabile `trattamento` l'antidoto. Come primo passo, si analizzano solamente i dati relativi al veleno II, creando il nuovo *data frame*

```
> TopiII <- subset(Topi, subset = veleno == "II",
+     select = -veleno)
> attach(TopiII)
```

Si vuole verificare se esiste differenza dei tempi medi di sopravvivenza per i 4 antidoti Si procede ad un'analisi grafica preliminare, anche per controllare le ipotesi di normalità e di omoschedasticità

```
> plot(trattamento, tempo)
```

Tenendo conto che si hanno solo 4 osservazioni per tipo di antidoto, appare difficile, dai diagrammi nella Figura 9.2, mettere in discussione l'assunzione di normalità. Sembra tuttavia esserci una diversità nella varianza dei vari gruppi.
Sia y_{ij} l'i-esima osservazione del j-esimo gruppo, con $i = 1, \ldots, 4$ e $j = 1, \ldots, 4$, dove $j = 1$ corrisponde all'antidoto A, $j = 2$ corrisponde all'antidoto B, $j = 3$ corrisponde all'antidoto C, $j = 4$ corrisponde all'antidoto D. Si assume allora che le y_{ij} siano realizzazioni di v.c. indipendenti $Y_{ij} \sim N(\mu_j, \sigma_j^2)$, per $i = 1, \ldots, 4$, $j = 1, \ldots, 4$.
Si desidera verificare l'ipotesi di omoschedasticità

$$H_0 \, : \, \sigma_1^2 = \sigma_2^2 = \sigma_3^2 = \sigma_4^2 \, .$$

attraverso il test di Bartlett

```
> bartlett.test(tempo, trattamento)

        Bartlett test for homogeneity of variances

data:  tempo and trattamento
Bartlett's K-squared = 9.5432, df = 3, p-value =
0.02288
```

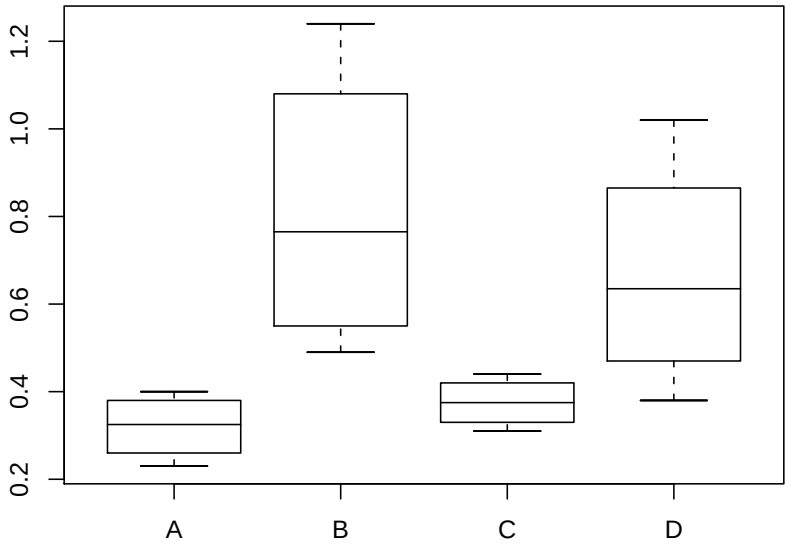


Figura 9.2: Diagrammi a scatola del tempo di sopravvivenza per i quattro antidoti.

Il livello di significatività osservato mostra una non trascurabile evidenza contro l'ipotesi nulla di uguaglianza delle varianze.
Si può provare a trasformare la variabile risposta, considerando, ad esempio, il reciproco del tempo di sopravvivenza

```
> plot(trattamento, 1/tempo)
```

La situazione sembra migliorata. Si assume dunque $1/Y_{ij} \sim N(\eta_j, \xi_j^2)$, per $i = 1, \ldots, 4$, $j = 1, \ldots, 4$. L'ipotesi di omoschedasticità relativa alle variabili trasformate è

$$H_0 \, : \, \xi_1^2 = \xi_2^2 = \xi_3^2 = \xi_4^2 \, ,$$

verificabile tramite il test di Bartlett

```
> bartlett.test(1/tempo, trattamento)

        Bartlett test for homogeneity of variances

data:  1/tempo and trattamento
Bartlett's K-squared = 1.2807, df = 3, p-value =
0.7337
```

In effetti le varianze non risultano ora significativamente diverse. Quindi si può pensare di verificare l'uguale efficacia dei 4 antidoti con riferimento alle medie dei

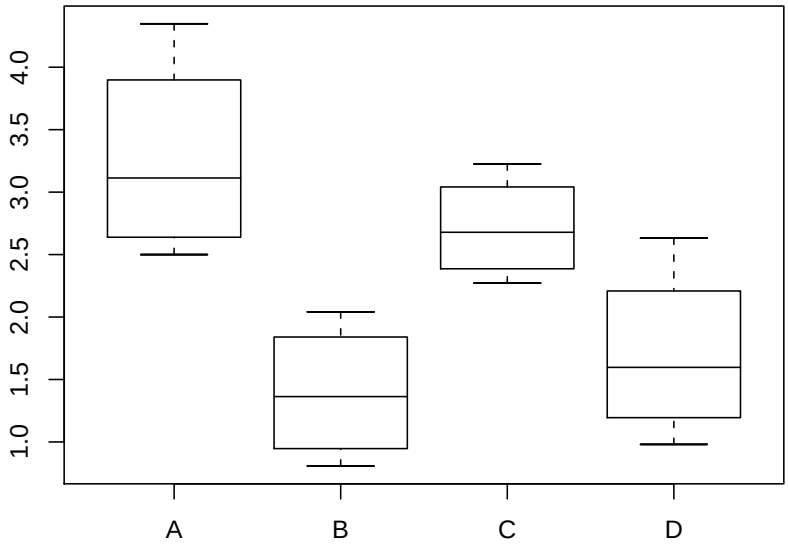


Figura 9.3: Diagrammi a scatola del reciproco del tempo di sopravvivenza per i quattro antidoti.

reciproci dei tempi di sopravvivenza. Assunta la normalità e l'omoschedasticità delle v.c. $1/Y_{ij}$, l'ipotesi di omogeneità

$$H_0 : \eta_1 = \eta_2 = \eta_3 = \eta_4$$

si può verificare con l'analisi della varianza

```
> topiII.aov <- aov(1/tempo ~ trattamento)
> summary(topiII.aov)

            Df Sum Sq Mean Sq F value   Pr(>F)
trattamento  3 9.1424  3.0475  7.3913 0.004594 **
Residuals   12 4.9477  0.4123
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1
```

I risultati evidenziano una diversità nelle medie dei reciproci dei tempi di sopravvivenza per i diversi antidoti.

9.3 Analisi dei dati MORLEY.DAT

I dati riportati nel file `Morley.dat` rappresentano misurazioni della velocità della luce nell'aria fatte tra il 5/6/1879 e il 2/7/1879 da Michelson e Morley. I dati sono

stati raccolti in 5 esperimenti (numerati da 1 a 5), ognuno dei quali consisteva di 20 misurazioni della velocità della luce nell'aria (in km/s). Si desidera valutare se esistono differenze significative nelle medie delle 5 popolazioni da cui provengono i dati, ossia dei 5 diversi esperimenti.
Si acquisiscano i dati

```
> Luce <- read.table("Morley.dat", header = TRUE)
```

I dati sono organizzati sotto forma di matrice dei dati. La variabile `Expt` indica l'esperimento, `Run` indica la misurazione per ciascun esperimento e `Speed` indica il valore osservato di velocità della luce.
È opportuno trasformare la variabile `Expt` in un fattore, visto che è solamente un'etichetta che individua i 5 esperimenti, e non una variabile quantitativa.

```
> Luce$Expt <- factor(Luce$Expt)
> attach(Luce)
```

Si procede con un'analisi preliminare dei dati. Volendo effettuare un'analisi della varianza è opportuno esplorare l'ipotesi di normalità ed omoschedasticità della distribuzione della velocità all'interno dei 5 gruppi definiti dai 5 esperimenti.

```
> plot(Speed ~ Expt)
```

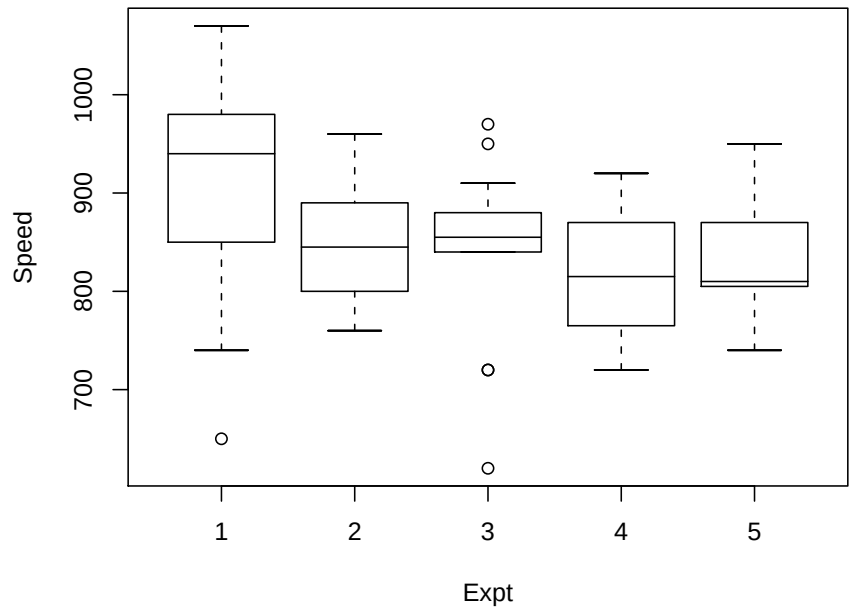


Figura 9.4: Diagrammi a scatola della velocità nei cinque esperimenti.

Basandosi solo sulla valutazione dei grafici nella Figura 9.4, l'ipotesi di omoschedasticità sembra discutibile.

Circa la normalità, si può dire che i gruppi 2 e 4 presentano una distribuzione sufficientemente simmetrica, mentre gli altri gruppi, in particolare il 3 ed il 5, evidenziano un'asimmetria della distribuzione. Per saggiare graficamente l'ipotesi di normalità si considerino i grafici quantile-quantile

```
> par(mfrow = c(2, 3), pty = "s")
> qqnorm(Speed[Expt == 1])
> qqline(Speed[Expt == 1])
> qqnorm(Speed[Expt == 2])
> qqline(Speed[Expt == 2])
> qqnorm(Speed[Expt == 3])
> qqline(Speed[Expt == 3])
> qqnorm(Speed[Expt == 4])
> qqline(Speed[Expt == 4])
> qqnorm(Speed[Expt == 5])
> qqline(Speed[Expt == 5])
```

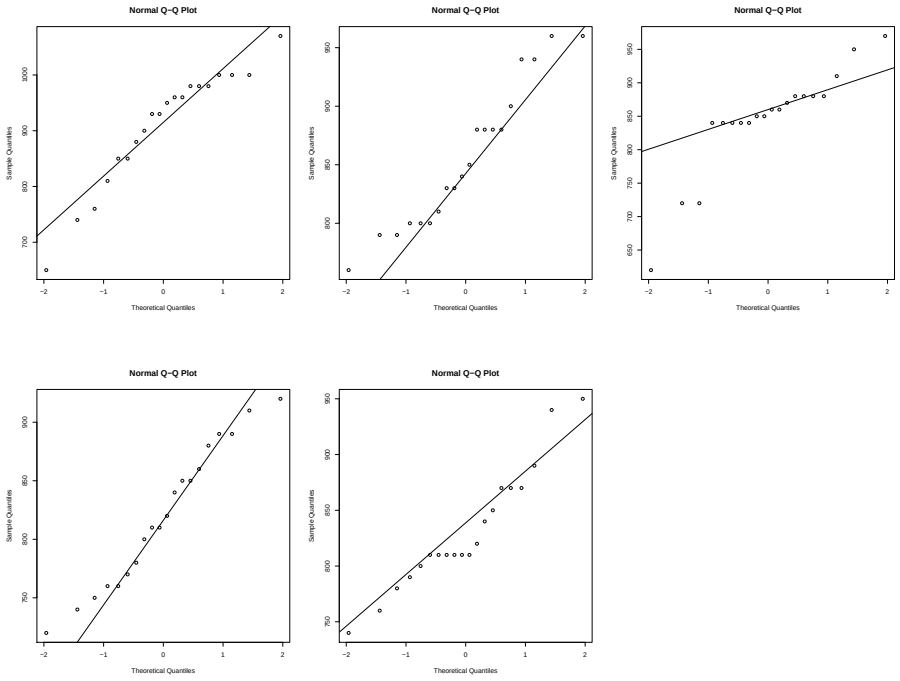


Figura 9.5: *Q-q plot* della velocità nei cinque esperimenti.

Anche tali diagrammi indicano che l'assunzione di normalità è ragionevole per i gruppi 2 e 4, mentre ci sono delle anomalie negli altri 3 gruppi.
L'allontanamento dalla normalità appare particolarmente marcato nel terzo gruppo. Si può ottenere qualche indicazione ulteriore relativa al terzo gruppo.

```
> summary(Speed[Expt == 3])
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    620     840     855     845     880     970
```

```
> sort(Speed[Expt == 3])
```

```
 [1] 620 720 720 840 840 840 840 840 850 850 860 860 870 880
[15] 880 880 880 910 950 970
```

```
> hist(Speed[Expt == 3])
```

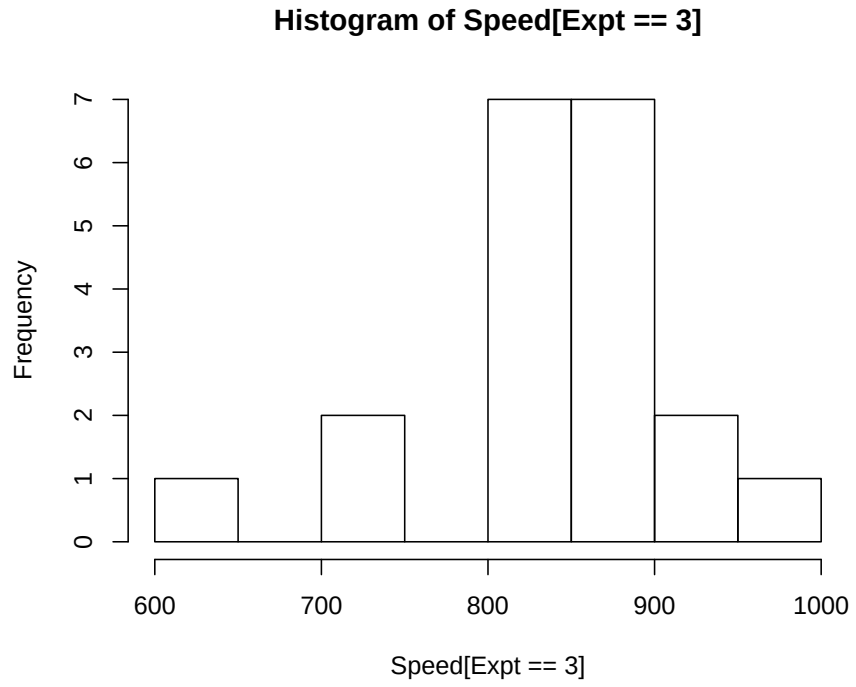


Figura 9.6: Istogramma della velocità nel terzo esperimento.

In pratica, le prime tre osservazioni risultano anomale (*outliers*) e ciò si osserva anche nel diagramma a scatola e nel *qq-plot* visti in precedenza.
Per il momento si procede ignorando il problema della non perfetta normalità e della possibile eteroschedasticità. Si indichino dunque con y_{ij} le osservazioni sulla velocità della luce, dove l'indice j, $j = 1, \ldots, 5$, segue i 5 esperimenti e l'indice i, $i = 1, \ldots, 20$, segue le 20 osservazioni per ciascun esperimento. Si assume che i valori y_{ij} siano realizzazioni di v.c. indipendenti $Y_{ij} \sim N(\mu_j, \sigma^2)$. L'ipotesi di omogeneità $H_0 : \mu_1 = \ldots = \mu_5$ può essere verificata tramite

```
> Luce.aov <- aov(Speed ~ Expt)
> summary(Luce.aov)
```

```
            Df Sum Sq Mean Sq F value   Pr(>F)
Expt         4  94514   23629  4.2878 0.003114 **
```

```
Residuals   95 523510    5511
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1
```

Sulla base delle assunzioni precedenti, l'evidenza empirica è dunque contraria ad H_0.

Esercizio. Si ripeta l'analisi della varianza utilizzando il comando `lm`. ◇

Per provare a risolvere il problema del gruppo 3, si può ripetere l'analisi dopo aver eliminato i valori anomali che corrispondo alle osservazioni più piccole.

```
> sort(Speed[Expt == 3])

 [1] 620 720 720 840 840 840 840 840 850 850 860 860 870 880
[15] 880 880 880 910 950 970
```

Si può creare un nuovo *dataframe* in cui non siano presenti le tre osservazioni anomale

```
> Luce1 = subset(Luce, !((Expt == 3) & (Speed <=
+     720)))
> detach()
> attach(Luce1)

> plot(Speed ~ Expt)
> hist(Speed[Expt == 3])
```

I diagrammi nella Figura 9.7 mostrano che la situazione rimane anomala. Si noti, d'altra parte, da

```
> summary(Speed[Expt == 3])

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  840.0   840.0   860.0   872.9   880.0   970.0

> sort(Speed[Expt == 3])

 [1] 840 840 840 840 840 850 850 860 860 870 880 880 880 880
[15] 910 950 970
```

che il primo quartile coincide con il minimo. Questo è dovuto al fatto che il valore minimo è stato osservato per ben 5 volte, ossia più del 25% delle osservazioni coincidono con tale valore. La distribuzione appare asimmetrica e l'ipotesi di normalità pare comunque forzata.
Quando le assunzioni alla base dell'analisi della varianza non sono soddisfatte, è possibile ricorrere ad altri test per saggiare l'ipotesi di omogeneità. In questo caso, ad esempio, si potrebbe utilizzare il test non parametrico di Kruskal-Wallis, utile per verificare l'ipotesi nulla di uguaglianza dei parametri di posizione in diverse popolazioni di cui non si specifica la distribuzione

```
> kruskal.test(Speed, Expt)
```

```
        Kruskal-Wallis rank sum test

data:  Speed and Expt
Kruskal-Wallis chi-squared = 17.7582, df = 4,
p-value = 0.001376
```

L'ipotesi nulla di uguaglianza dei parametri di posizione delle distribuzioni viene rifiutata, con un livello di significatività osservato prossimo a quello calcolato tramite l'analisi della varianza.

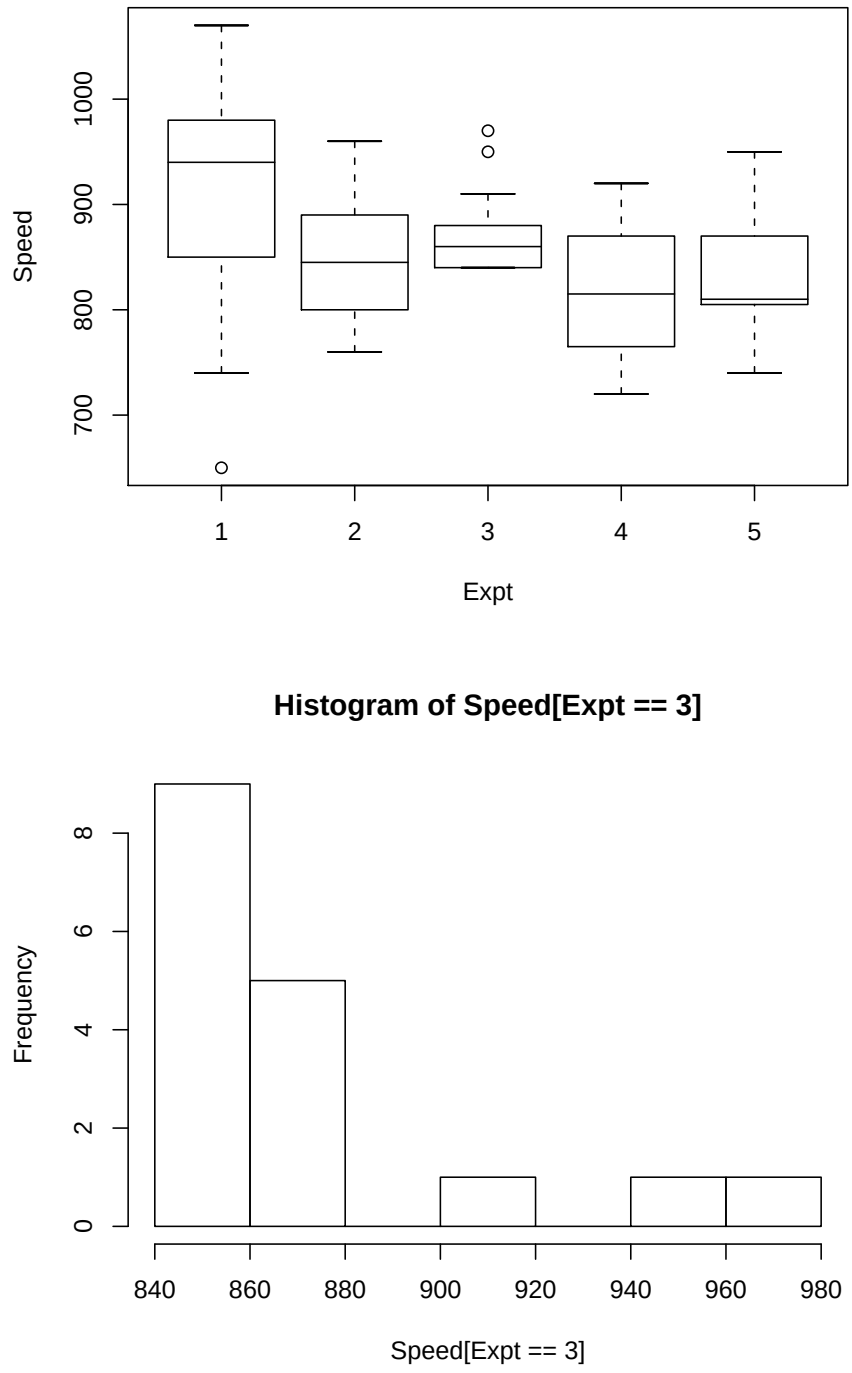


Figura 9.7: Diagrammi a scatola della velocità nei cinque esperimenti, dopo aver eliminato le tre osservazioni più piccole del terzo esperimento.

Capitolo 10

Analisi della varianza a due fattori

10.1 Analisi dei dati PENICILLIN.DAT

I dati contenuti nel file `penicillin.dat`, si riferiscono ad un esperimento volto a confrontare la produttività di 4 processi (A,B,C,D) di sintesi industriale di penicillina. Tali processi possono essere basati sull'uso di diversi fermentatori, ovvero terreni di coltura capaci di incrementare la produttività. Nell'esperimento considerato, vengono utilizzati 5 fermentatori (I,II,III,IV,V). Per ogni processo, si dispone della quantità di penicillina prodotta per ciascuno dei 5 fermentatori. In sintesi, i dati hanno il seguente aspetto.

	fermentatore				
processo	I	II	III	IV	V
A	89	84	81	87	79
B	88	77	87	92	81
C	97	92	87	89	80
D	94	79	85	84	88

Si è di fronte, quindi, ad un esempio di esperimento fattoriale 5×4 con una replicazione, nel senso che si ha una sola osservazione sulla variabile risposta (la produttività) per ciascuno dei 20 incroci fermentatore-processo. Interessa sondare se esiste una diversità nella quantità media di penicillina prodotta nei 4 processi, indipendentemente dal fermentatore utilizzato.
Si acquisiscano i dati e si proceda con un'analisi esplorativa.

```
> Pen <- read.table("penicillin.dat", header = TRUE)
> attach(Pen)
```

Si noti che i dati sono già organizzati sotto forma di matrice dei dati. La variabile `modo` rappresenta il processo produttivo e la variabile `miscela` il fermentatore.

```
> plot(penicillina ~ modo)
> plot(penicillina ~ miscela)
```

I grafici nella Figura 10.1 evidenziano delle differenze (in termini di mediane) sia tra i processi che tra i fermentatori, anche se si deve tener conto della ridotta numerosità

campionaria. La variabilità entro i gruppi sembrerebbe comparabile. Si nota una certa asimmetria nelle distribuzioni.
Si consideri in primo luogo l'effetto del processo produttivo, senza tener conto del possibile effetto del fermentatore. Si sviluppi quindi una analisi della varianza ad un fattore (`modo`).

```
> pen.aov <- aov(penicillina ~ modo)
> summary(pen.aov)

            Df Sum Sq Mean Sq F value Pr(>F)
modo         3  70.00   23.33  0.7619 0.5318
Residuals   16 490.00   30.62
```

Il livello di significatività osservato del test di uguaglianza dei livelli medi di produzione nei 4 processi porta all'accettazione dell'ipotesi nulla, contrariamente a quanto suggerito dall'analisi esplorativa. Tale risultato potrebbe essere influenzato dalla presenza di diversi fermentatori. Si provi quindi a tenere conto di questo, considerando dapprima l'effetto marginale dei fermentatori.

```
> pen.aov.1 <- aov(penicillina ~ miscela)
> summary(pen.aov.1)

            Df  Sum Sq Mean Sq F value  Pr(>F)
miscela      4 264.000  66.000  3.3446 0.03801 *
Residuals   15 296.000  19.733
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1
```

Si può osservare come il fermentatore influenzi, in media, la produzione.

Si consideri ora il modello di analisi della varianza a due fattori. Si utilizzi, per rappresentare i valori della risposta, la notazione y_{ijk}, dove l'indice $j,\ j = 1, \ldots, 4$, indicizza i 4 processi, l'indice $k,\ k = 1, \ldots, 5$, indicizza i 5 fermentatori e l'indice i indicizza l'osservazione all'interno del gruppo jk. Avendo ciascun gruppo jk una sola osservazione, l'indice i assume per ogni gruppo solo il valore 1; nel seguito, pertanto, tale indice verrà omesso per semplicità. Siano quindi $y_{jk},\ j = 1, \ldots, 4,\ k = 1, \ldots, 5$, i valori della risposta. Si consideri il modello statistico che ipotizza che le y_{jk} siano realizzazioni di v.c. Y_{jk} tali che:

$$Y_{jk} = \mu + \alpha_j + \gamma_k + \varepsilon_{jk},$$

$\varepsilon_{jk} \sim N(0, \sigma^2)$ indipendenti, per $j = 1, \ldots, 4$ e $k = 1, \ldots, 5$. Si proceda quindi alla analisi.

```
> pen.aov.2 <- aov(penicillina ~ modo + miscela)
> summary(pen.aov.2)
```

```
            Df  Sum Sq Mean Sq F value  Pr(>F)
modo         3  70.000  23.333  1.2389 0.33866
miscela      4 264.000  66.000  3.5044 0.04075 *
Residuals   12 226.000  18.833
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1
```

La funzione `aov()` gestisce automaticamente il problema della non identificabilità dei parametri del modello sopra specificato, ponendo $\alpha_1 = 0$ e $\gamma_1 = 0$. Si osservi come il fermentatore abbia un effetto leggermente significativo sulla produzione; i diversi processi, invece, sembrano non influire significativamente, in media, sulla produzione di penicillina.

10.2 Analisi dei dati RATS.DAT

I dati contenuti nel file `rats.dat`, si riferiscono ad un disegno fattoriale 3×4 completamente randomizzato sull'effetto di agenti tossici ed antidoti. Si considerano 3 tipi di veleno (I, II, III) e 4 antidoti (A, B, C, D). Ogni combinazione veleno-antidoto viene sperimentata su 4 cavie. Per ogni cavia, viene osservato il tempo di sopravvivenza espresso in decine di ore. I dati, pertanto, risultano così costruiti.

Antidoto	Veleno				
A	I	0.31	0.45	0.46	0.43
	II	0.36	0.29	0.40	0.23
	III	0.22	0.21	0.18	0.23
B	I	0.82	1.10	0.88	0.72
	II	0.92	0.61	0.49	1.24
	III	0.30	0.37	0.38	0.29
C	I	0.43	0.45	0.63	0.76
	II	0.44	0.35	0.31	0.40
	III	0.23	0.25	0.24	0.22
D	I	0.45	0.71	0.66	0.62
	II	0.56	1.02	0.71	0.38
	III	0.30	0.36	0.31	0.33

```
> Topi <- read.table("rats.dat", header = TRUE)
> attach(Topi)
```

Si noti che i dati sono già organizzati sotto forma di matrice dei dati. La variabile `veleno` rappresenta il veleno somministrato e la variabile `trattamento` l'antidoto. Si parte con l'analizzare le distribuzioni del tempo di sopravvivenza marginalmente rispetto ad ognuno dei due fattori.

```
> plot(tempo ~ veleno)
> plot(tempo ~ trattamento)
```

Dai grafici riportati nella Figura 10.2 si può vedere che le distribuzioni non appaiono simmetriche e che, in generale, i fattori hanno un effetto sulla risposta. Con i comandi seguenti si può valutare graficamente l'effetto di interazione.

```
> interaction.plot(veleno, trattamento, tempo)
> interaction.plot(trattamento, veleno, tempo)
```

I grafici ottenuti, detti diagrammi di interazione, sono riportati nella Figura 10.3. Se le spezzate di un diagramma di interazione appaiono abbastanza parallele vi è un'indicazione di non interazione. Non vi è interazione se le differenze tra le medie della risposta per diversi livelli del primo fattore sono influenzati dai livelli del secodno fattore (o viceversa). Nel caso in esame è possibile sottoporre a verifica l'ipotesi di interazione, visto che si dispone di più di una osservazione per ogni combinazione dei fattori. Ciò non era possibile nell'esempio del paragrafo 10.1. Siano y_{ijk}, $i = 1, \ldots, n_{jk}$, $j = 1, \ldots, 3$, $k = 1, \ldots, 4$, i valori della risposta. Nell'esempio considerato, è $n_{jk} = 4$ per ogni j, k. Si consideri il modello statistico che ipotizza che le y_{ijk} siano realizzazioni di v.c. Y_{ijk} tali che:

$$Y_{ijk} = \mu + \alpha_j + \gamma_k + \delta_{jk} + \varepsilon_{ijk}.$$

$\varepsilon_{ijk} \sim N(0, \sigma^2)$ indipendenti, per $i = 1, \ldots, 4$, $j = 1, \ldots, 3$ e $k = 1, \ldots, 4$. Il modello è adattabile utilizzando la funzione `aov()`. Si faccia attenzione alla sintassi della formula nel comando.

```
> topi.aov <- aov(tempo ~ veleno * trattamento)
> summary(topi.aov)
```

```
                   Df  Sum Sq Mean Sq F value    Pr(>F)
veleno              2 1.03301 0.51651 23.2217 3.331e-07 ***
trattamento         3 0.92121 0.30707 13.8056 3.777e-06 ***
veleno:trattamento  6 0.25014 0.04169  1.8743    0.1123
Residuals          36 0.80072 0.02224
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1
```

Come si può osservare, l'effetto di interazione (`veleno:trattamento`) non è statisticamente significativo.

Alla stessa conclusione si sarebbe giunti utilizzando la funzione `lm()` e confrontando mediante il test F due modelli di regressione, il modello completo, che prevede gli effetti marginali dei fattori e la loro interazione, ed il modello ridotto, che prevede solo gli effetti marginali. Si faccia nuovamente attenzione alla sintassi della formula nel modello completo.

```
> topi.lm.ridotto <- lm(tempo ~ veleno + trattamento)
> topi.lm.completo <- lm(tempo ~ veleno * trattamento)
> anova(topi.lm.ridotto, topi.lm.completo)
```

```
Analysis of Variance Table

Model 1: tempo ~ veleno + trattamento
Model 2: tempo ~ veleno * trattamento
  Res.Df     RSS Df Sum of Sq      F Pr(>F)
1     42 1.05086
2     36 0.80072  6   0.25014 1.8743 0.1123
```

La funzione `anova()` sviluppa il seguente test.

```
> n <- dim(Topi)[1]
> n

[1] 48

> sigma2.tilde <- sum(topi.lm.ridotto$resid^2)/n
> sigma2.cappello <- sum(topi.lm.completo$resid^2)/n
> gdl.ridotto <- topi.lm.ridotto$df.resid
> gdl.completo <- topi.lm.completo$df.resid
> p0 <- n - gdl.ridotto
> p <- n - gdl.completo
> p - p0

[1] 6

> n - p

[1] 36

> F.test <- ((sigma2.tilde - sigma2.cappello)/(p -
+     p0))/(sigma2.cappello/(n - p))
> F.test

[1] 1.874333

> 1 - pf(F.test, p - p0, n - p)

[1] 0.1122506
```

La scomposizione della varianza contenuta nell'oggetto `topi.aov` è reperibile anche mediante il seguente comando.

```
> anova(topi.lm.completo)

Analysis of Variance Table

Response: tempo
                   Df  Sum Sq Mean Sq F value    Pr(>F)
veleno              2 1.03301 0.51651 23.2217 3.331e-07 ***
trattamento         3 0.92121 0.30707 13.8056 3.777e-06 ***
veleno:trattamento  6 0.25014 0.04169  1.8743    0.1123
Residuals          36 0.80072 0.02224
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1
```

Si considerino ora i residui.

```
> topi.rstand <- rstandard(topi.lm.completo)
> qqnorm(topi.rstand)
> qqline(topi.rstand)
> plot(fitted(topi.lm.completo), topi.rstand)
```

L'ipotesi di omoschedasticità non sembra suffragata. Si provi a trasformare i dati logaritmicamente.

```
> topi.lm.1 <- lm(log(tempo) ~ veleno * trattamento)
> plot(fitted(topi.lm.1), rstandard(topi.lm.1))
```

La Figura 10.5 indica che la trasformazione logaritmica non sembra portare alcun miglioramento. Si provi allora a considerare il reciproco del tempo di sovravvivenza.

```
> topi.lm.2 <- lm(1/tempo ~ veleno * trattamento)
> plot(fitted(topi.lm.2), rstandard(topi.lm.2))
```

La Figura 10.6 indica che l'ultimo modello sembra catturare meglio la variabilità della risposta.
Si consideri l'analisi della normalità dei residui.

```
> qqnorm(rstandard(topi.lm.2))
> qqline(rstandard(topi.lm.2))
```

Anche il grafico quantile-quantile è migliorato, si veda la Figura 10.7. Si ricordi che la trasformazione adottata tramite la funzione reciproco era suggerita anche nel Paragrafo 9.2. Si effettui allora un'analisi della varianza con tale trasformazione della variabile risposta.

```
> anova(topi.lm.2)

Analysis of Variance Table

Response: 1/tempo
                  Df Sum Sq Mean Sq F value    Pr(>F)
veleno             2 34.877  17.439 72.6347 2.310e-13 ***
trattamento        3 20.414   6.805 28.3431 1.376e-09 ***
veleno:trattamento 6  1.571   0.262  1.0904    0.3867
Residuals         36  8.643   0.240
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1
```

L'analisi della varianza non mostra alcuna indicazione della presenza di interazione. Si può stimare un modello senza interazione.

```
> topi.lm.3 <- lm(1/tempo ~ veleno + trattamento)
> anova(topi.lm.3)
```

```
Analysis of Variance Table

Response: 1/tempo
            Df Sum Sq Mean Sq F value    Pr(>F)
veleno       2 34.877  17.439  71.708 2.865e-14 ***
trattamento  3 20.414   6.805  27.982 4.192e-10 ***
Residuals   42 10.214   0.243
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1
```

```
> qqnorm(rstandard(topi.lm.3))
> qqline(rstandard(topi.lm.3))
```

Esercizio. I dati contenuti nel file `nails.dat`, si riferiscono ad un esperimento per mezzo del quale si vuole valutare l'efficacia di 3 smacchiatori nello sciogliere dai tessuti macchie di 3 tipi di smalto per unghie. L'esperimento consiste nell'immergere in una bacinella con un certo solvente 5 tessuti macchiati da un certo tipo di smalto, misurando il tempo (in minuti) necessario al dissolvimento della macchia. Si valuti se esistono diversità di azione dei solventi, anche rispetto al tipo di macchia. ◇

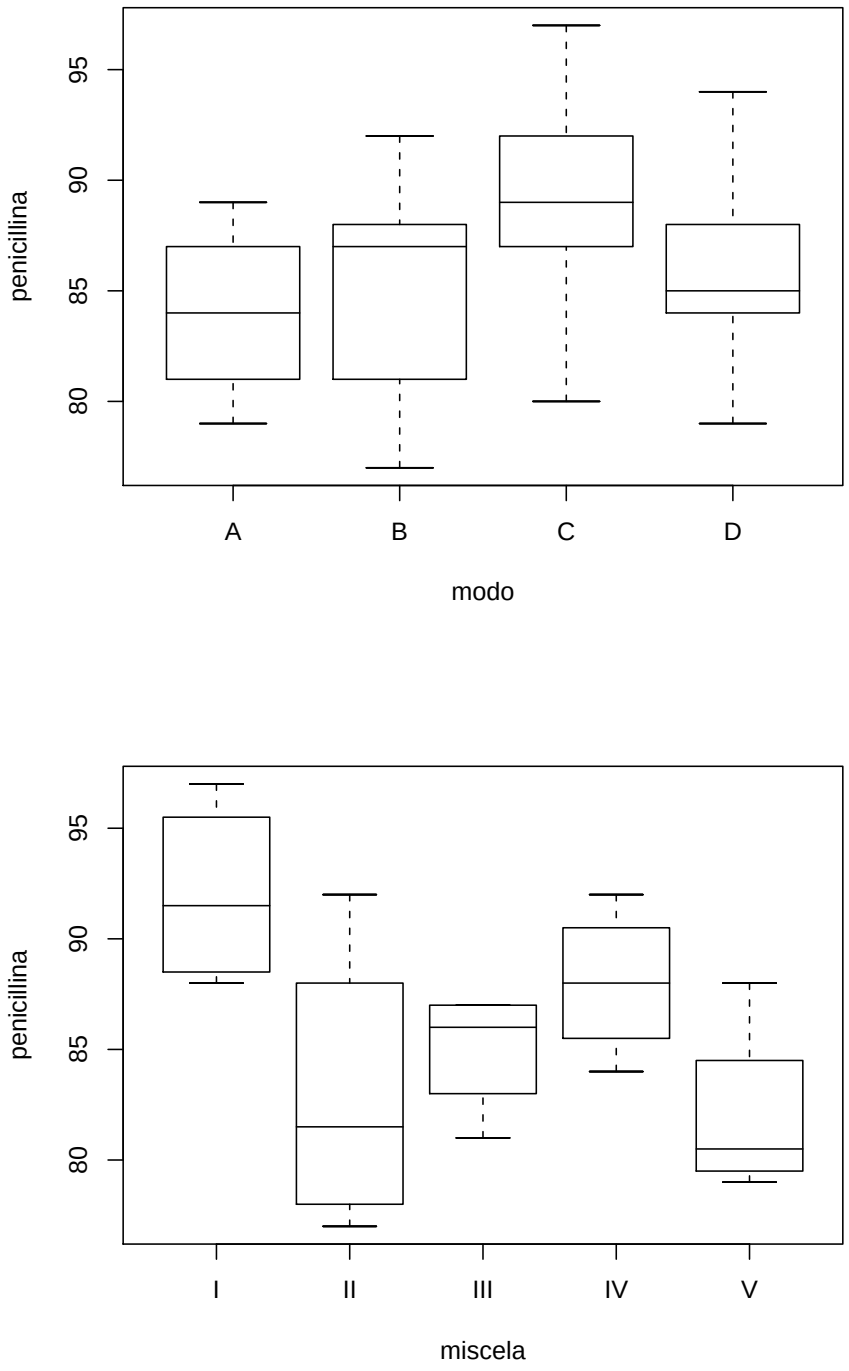


Figura 10.1: Diagrammi a scatola della quantità di penicillina nei quattro processi produttivi e con i cinque fermentatori.

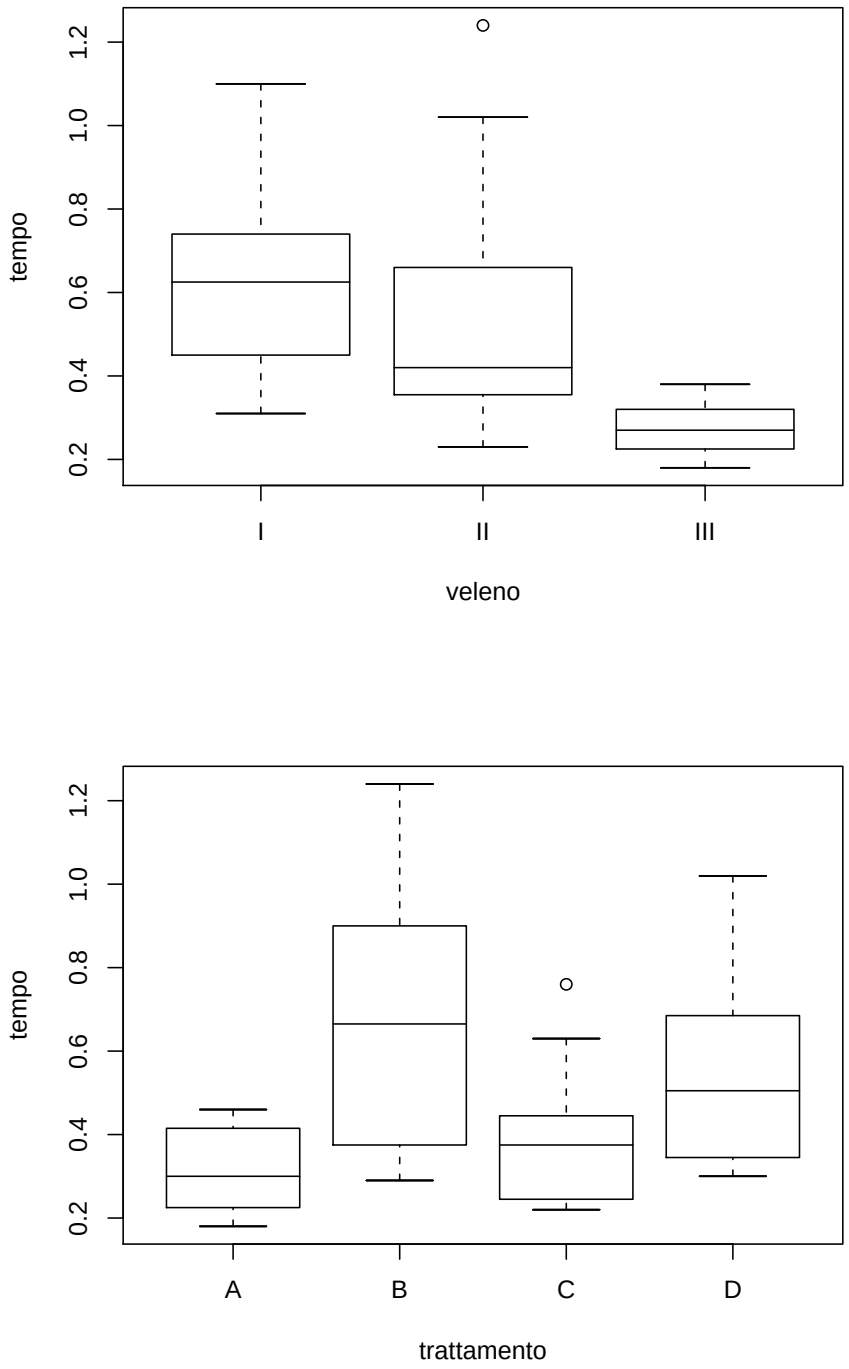


Figura 10.2: Diagrammi a scatola del tempo di sopravvivenza per i tre veleni e per i quattro antidoti.

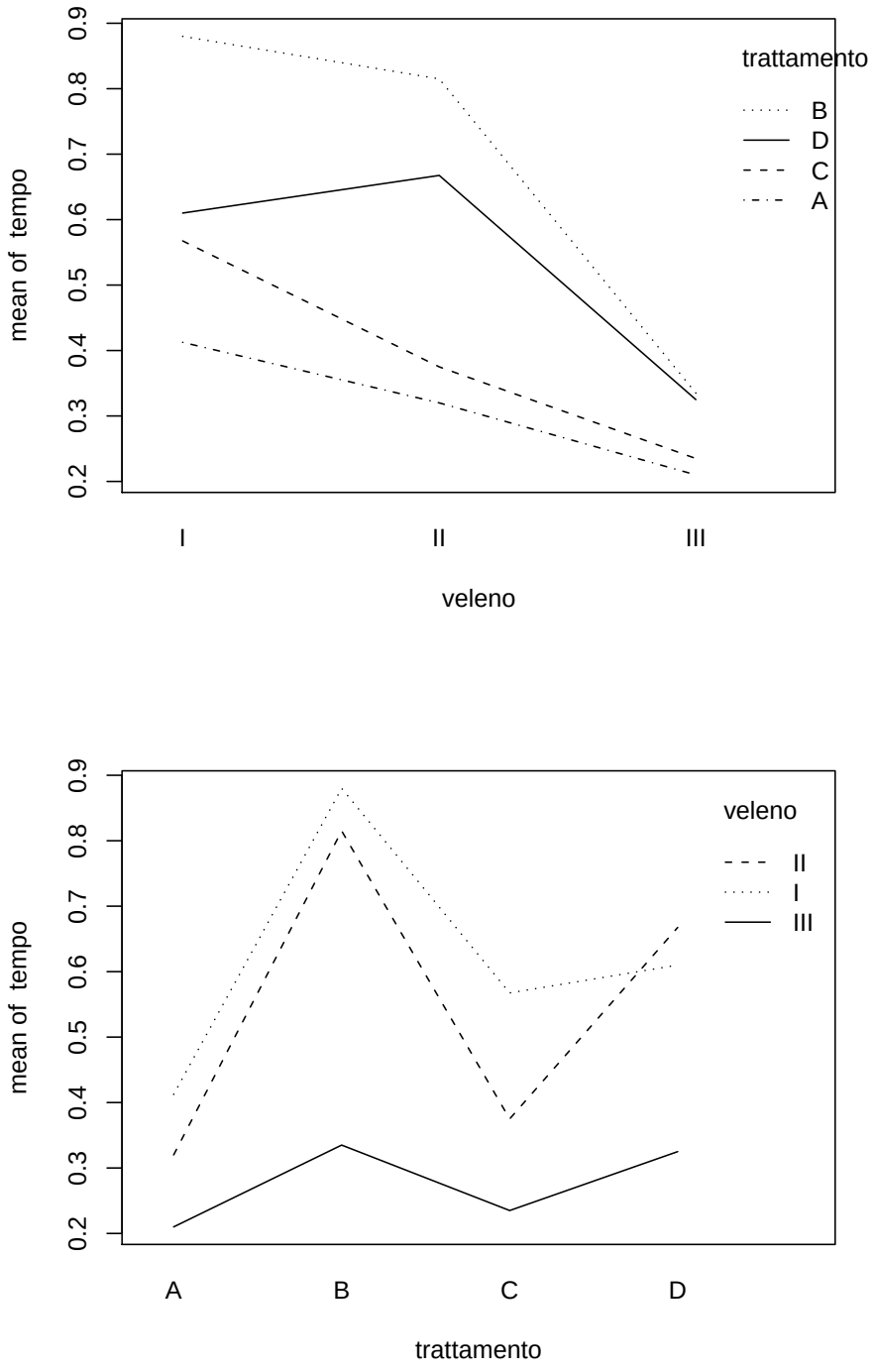

mean of tempo
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
I
II
III
veleno
trattamento
B
D
C
A
mean of tempo
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
A
B
C
D
trattamento
veleno
II
I
III

Figura 10.3: .

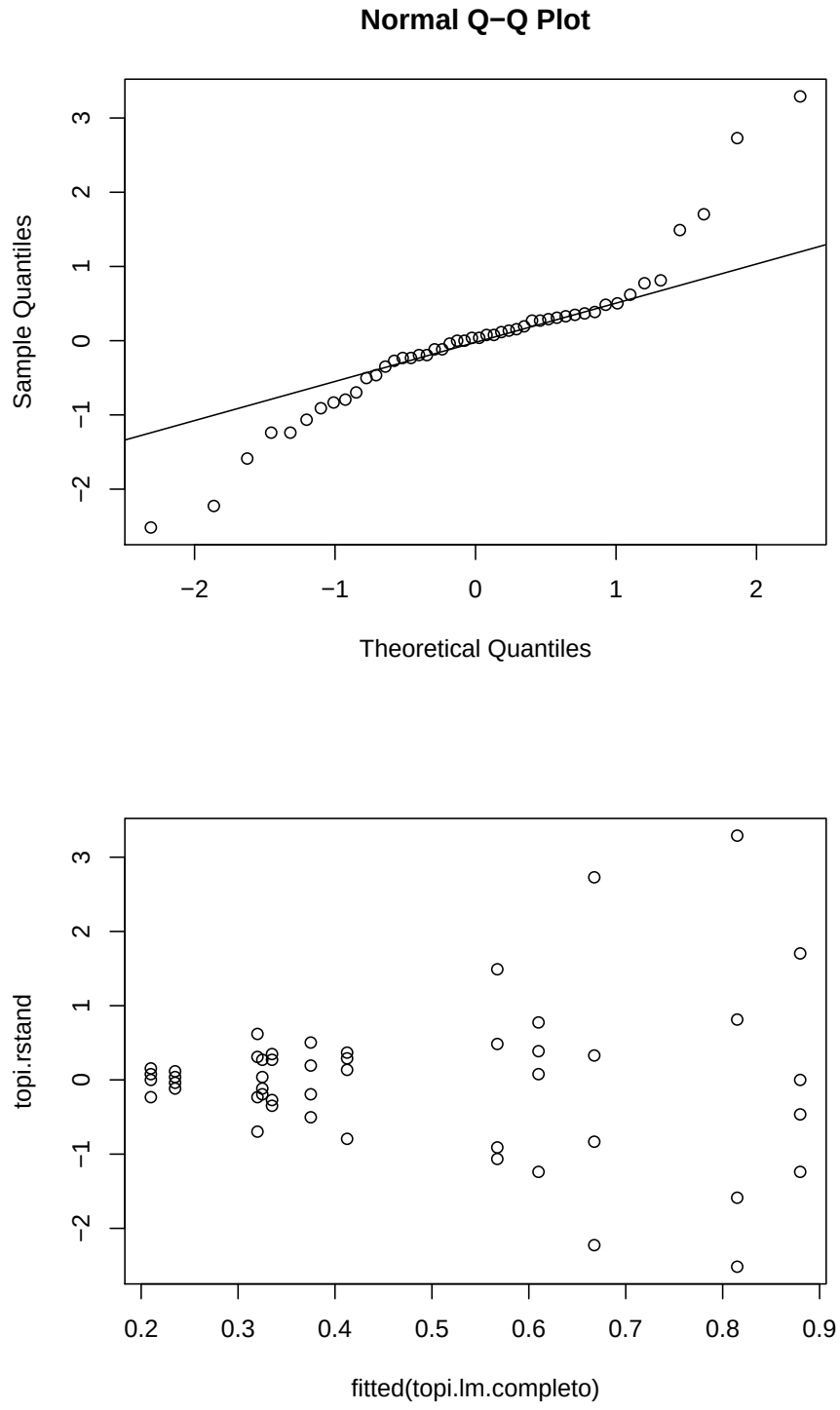


Figura 10.4: *Q-q plot* e diagramma di dispersione, rispetto ai valori predetti, dei residui standardizzati di `topi.lm.completo`.

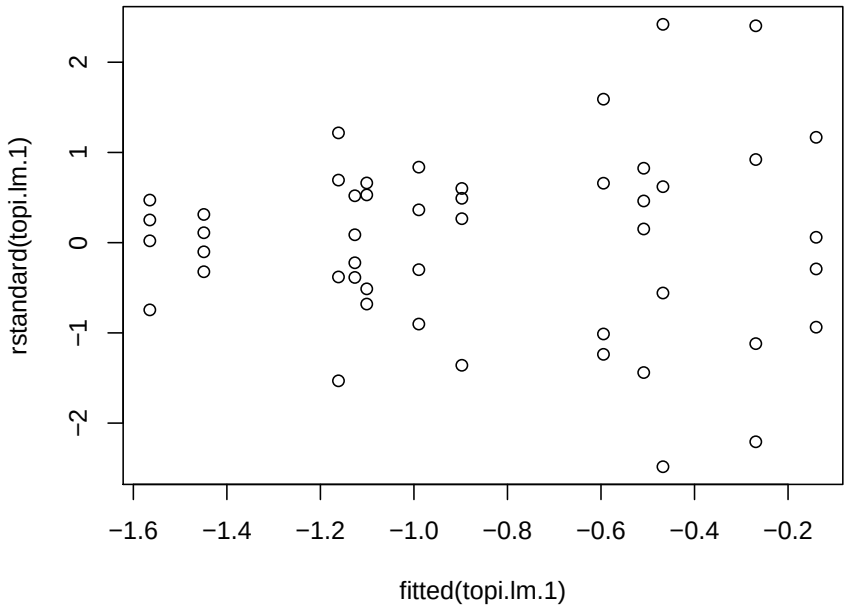


Figura 10.5: Diagramma di dispersione dei residui standardizzati di `topi.lm.1`.

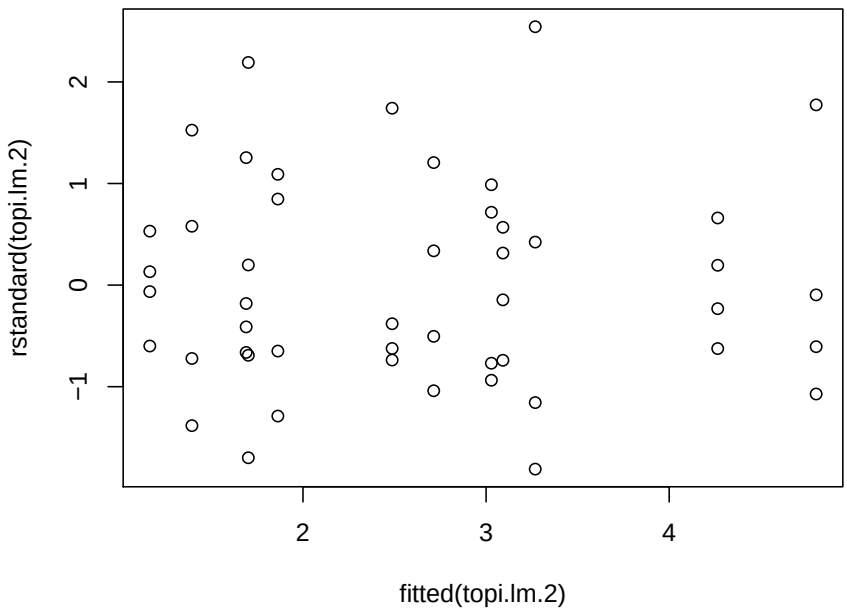


Figura 10.6: Diagramma di dispersione dei residui standardizzati di `topi.lm.2`.

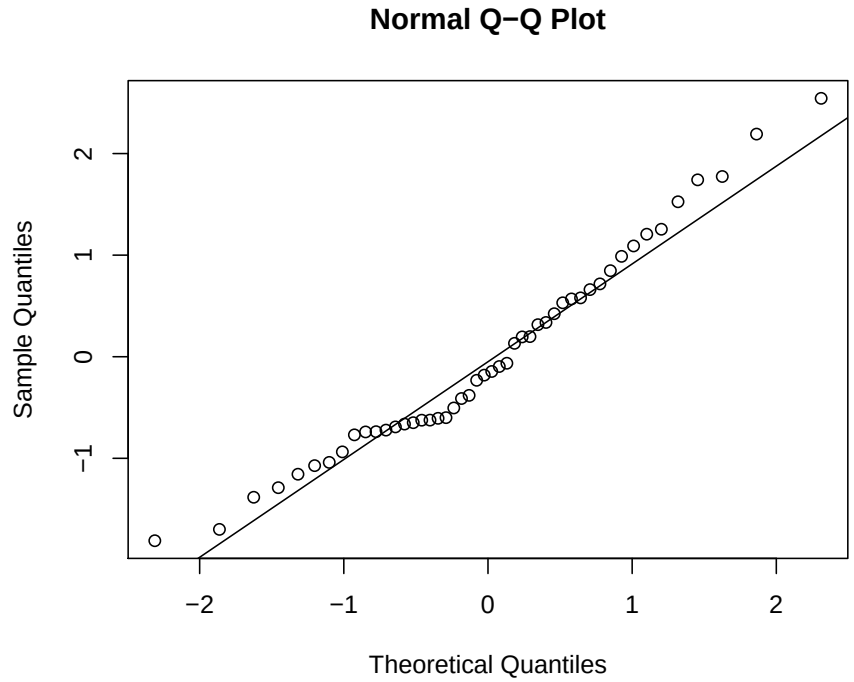


Figura 10.7: *Q-q plot* dei residui standardizzati di `topi.lm.2`.

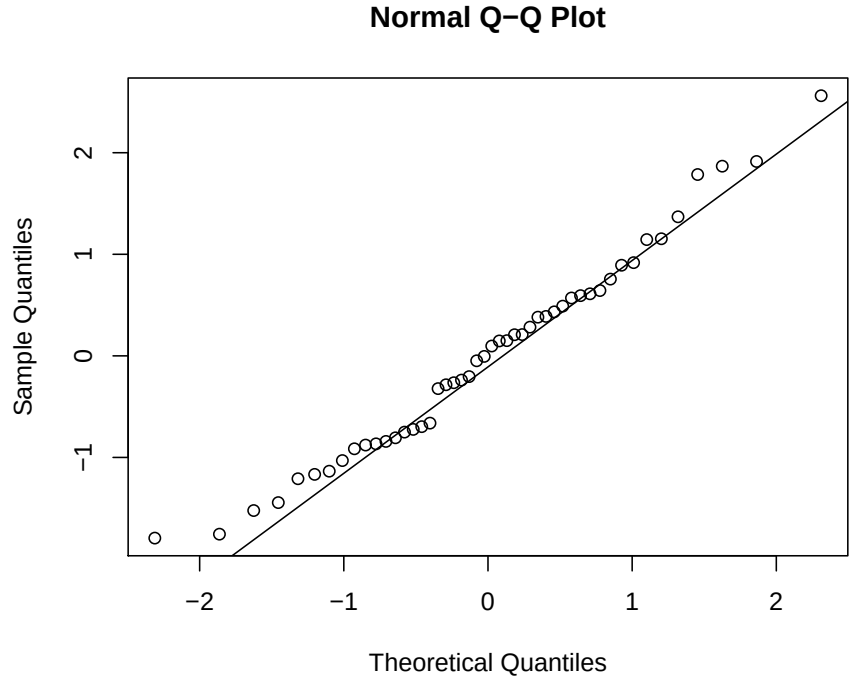


Figura 10.8: *Q-q plot* dei residui standardizzati di `topi.lm.3`.

Capitolo 11

Analisi della covarianza

11.1 Analisi dei dati CATS.DAT

I dati contenuti nel file `cats.dat` presentano il peso del corpo (in kg) ed il peso del cuore (in g) di alcuni gatti di sesso femminile (`1`) e maschile (`2`). Si vuole verificare se esistono differenze, in media, nel peso del cuore tra i due sessi, tenendo presente però la relazione che esiste tra peso del cuore e peso del corpo.

```
> Gatti <- read.table("cats.dat", col.names = c("pcorpo",
+     "pcuore", "sesso"))
```

Si noti che la variabile `sesso` non è automaticamente riconosciuta come variabile qualitativa, essendo codificata mediante i valori `1` e `2`. È pertanto necessario dichiararla come tale. Seppure non sia non obbligatorio, può essere conveniente, per facilitare la lettura delle analisi successive, cambiare le etichette alle modalità.

```
> Gatti$sesso[Gatti$sesso == 1] = "F"
> Gatti$sesso[Gatti$sesso == 2] = "M"
> Gatti$sesso <- factor(Gatti$sesso)
> attach(Gatti)
```

In via preliminare, si esamina la distribuzione del peso del cuore nei due sessi.

```
> plot(pcuore ~ sesso)
```

Il grafico nella Figura 11.1 mostra una chiara differenza tra i due sessi. Il peso del cuore nelle femmine è mediamente più basso. Se volessimo verificare l'esistenza di una differenza in media tra i due gruppi, potremmo utilizzare il test t di Student, previa verifica della normalità ed omoschedasticità delle due distribuzioni.

```
> qqnorm(pcuore[sesso == "F"])
> qqline(pcuore[sesso == "F"])
> qqnorm(pcuore[sesso == "M"])
> qqline(pcuore[sesso == "M"])
```

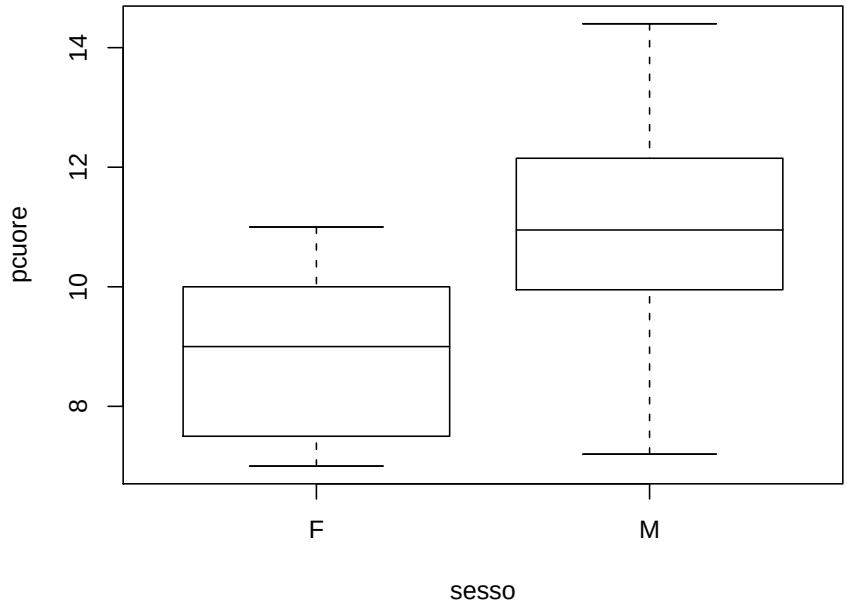


Figura 11.1: Diagrammi a scatola del peso del cuore per i due sessi.

I *qq-plot* nella Figura 11.2 indicano che la distribuzione del peso del cuore nelle femmine devia, sulle code, dalla normalità (questo era da attendersi visto il diagramma a scatola), mentre l'ipotesi di normalità pare più che accettabile per i maschi.

Per quanto riguarda l'omoschedasticità, è possibile condurre il test di omogeneità delle varianze.

```
> var.test(pcuore[sesso == "F"], pcuore[sesso ==
+     "M"])

        F test to compare two variances

data:  pcuore[sesso == "F"] and pcuore[sesso == "M"]
F = 0.4799, num df = 23, denom df = 23, p-value =
0.08496
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.2076113 1.1094088
sample estimates:
ratio of variances
         0.4799227
```

L'ipotesi nulla di uguaglianza delle varianze nelle due popolazioni non è rifiutata, per cui si può procedere con il test t, nonostante le incertezze sulla normalità della

distribuzione per le femmine. Vista la natura del confronto, si può utilizzare il test t con alternativa unilaterale

$$\begin{aligned} H_0 &: \mu_F = \mu_M \\ H_1 &: \mu_F < \mu_M \,, \end{aligned}$$

dove μ_F indica la media della popolazione dei gatti di sesso femminile e μ_M la media della popolazione dei gatti di sesso maschile. Gli esiti del test sono mostrati in quanto segue.

```
> t.test(pcuore[sesso == "F"], pcuore[sesso == "M"],
+     alternative = "less", var.equal = TRUE)

        Two Sample t-test

data:  pcuore[sesso == "F"] and pcuore[sesso == "M"]
t = -4.8419, df = 46, p-value = 7.455e-06
alternative hypothesis: true difference in means is less than 0
95 percent confidence interval:
      -Inf -1.412780
sample estimates:
mean of x mean of y
   8.8875   11.0500
```

Come era da attendersi, il test rifiuta l'ipotesi nulla H_0.

Nella formulazione del problema si chiedeva però di tenere conto nel confronto della relazione esistente tra peso del corpo e peso del cuore. Si provi quindi a esplorare la relazione esistente tra le due variabili.

```
> plot(pcuore ~ pcorpo)
```

Il grafico nella Figura 11.3 mostra una chiara relazione monotona crescente tra peso del cuore e del corpo, come è logico attendersi. Si provi a vedere la distribuzione del peso del corpo nei due sessi.

```
> plot(pcorpo ~ sesso)
```

Come immaginabile, la Figura 11.4 mostra che il peso del corpo dei gatti maschi è mediamente più alto di quello delle femmine. Considerata allora la relazione tra peso del corpo e quello del cuore, si può pensare che le differenze osservate nel peso medio del cuore tra i due sessi siano in realtà dovute alla dipendenza tra peso del corpo e quello del cuore, più che all'appartenenza al sesso.

```
> plot(pcuore ~ pcorpo, type = "n")
> points(pcorpo[sesso == "F"], pcuore[sesso == "F"],
+     pch = 1)
> points(pcorpo[sesso == "M"], pcuore[sesso == "M"],
+     pch = 1, col = 2)
```

Per verificare l'ipotesi prima formulata, si dovrebbe effettuare un confronto tra il peso medio del cuore di maschi e femmine al netto del peso del corpo. L'analisi della covarianza consente di sviluppare tale confronto.

A tal fine, si indichino con (y_i, x_i, z_i), $i = 1, \ldots, 48$, le 48 terne di valori contenenti nell'ordine, per ciascuna unità statistica, il peso del cuore, il peso del corpo e l'indicatore del sesso. Si supponga che la variabile indicatrice z_i, che codifica il sesso, assuma il valore 0 per le femmine e 1 per i maschi. Si assuma che $y_1, \ldots, y_{48}$ siano realizzazioni di v.c. indipendenti Y_i tali che

$$Y_i = \beta_1 + \beta_2\, x_i + \beta_3\, z_i + \beta_4\, x_i\, z_i + \varepsilon_i$$

con $\varepsilon_i \sim N(0, \sigma^2)$ indipendenti, $i = 1, \ldots, 48$. Tale modello prevede per $i = 1, \ldots, 24$, ossia per il sesso femminile, $Y_i \sim N(\beta_1 + \beta_2\, x_i, \sigma^2)$, mentre per $i = 25, \ldots, 48$, ossia per il sesso maschile, $Y_i \sim N((\beta_1 + \beta_3) + (\beta_2 + \beta_4)\, x_i, \sigma^2)$.

L'ipotesi nulla di uguaglianza del peso medio del cuore nei due sessi al netto dell'effetto del peso del corpo corrisponde all'ipotesi statistica

$$\begin{array}{ll} H_0\text{:} & \beta_3 = \beta_4 = 0 \\ H_1\text{:} & \bar{H}_0 \end{array}$$

La verifica di tale ipotesi può essere condotta utilizzando le funzioni `lm()` e `anova()`. Prima di procedere, si controlli come `R` ha codificato il fattore `sesso`.

```
> contrasts(sesso)

  M
F 0
M 1
```

Dunque, viene associato il valore 1 alla modalità `M` e 0 alla modalità `F`. La codifica effettuata da `R` coincide quindi con la definizione della variabile z_i utilizzata nel modello statistico sopra definito. Il modello viene adattato con

```
> gatti.lm <- lm(pcuore ~ pcorpo * sesso)
```

o, equivalentemente,

```
> gatti.lm <- lm(pcuore ~ pcorpo + sesso + pcorpo:sesso)
```

Il risultato dell'analisi è il seguente.

```
> summary(gatti.lm)

Call:
lm(formula = pcuore ~ pcorpo + sesso + pcorpo:sesso)

Residuals:
    Min      1Q  Median      3Q     Max
-1.9813 -0.9589 -0.1629  0.8573  2.6277
```

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)     2.9318     2.1105   1.389  0.17178
pcorpo          2.5525     0.8975   2.844  0.00674 **
sessoM         -0.2849     3.0313  -0.094  0.92554
pcorpo:sessoM   0.4177     1.1784   0.354  0.72466
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

Residual standard error: 1.28 on 44 degrees of freedom
Multiple R-Squared: 0.5664,        Adjusted R-squared: 0.5368
F-statistic: 19.16 on 3 and 44 DF,  p-value: 4.269e-08
```

I risultati dei test di nullità dei coefficienti di regressione suggeriscono che i coefficienti β_3 e β_4, singolarmente considerati, possono essere considerati nulli. In altre parole, le ipotesi $H_0 : \beta_3 = 0$ e $H_0 : \beta_4 = 0$ non vengono rifiutate. Questo sembrerebbe indicare che le relazioni lineari nei due sessi abbiano uguale intercetta e uguale coefficiente angolare.
Tuttavia, i due test non verificano l'ipotesi nulla $H_0 : \beta_3 = \beta_4 = 0$, che è l'oggetto di interesse. Per verificare tale ipotesi, dobbiamo confrontare il modello completo sopra definito con il modello ridotto

$$Y_i = \beta_1 + \beta_2 \, x_i + \varepsilon_i,$$

con $\varepsilon_i \sim N(0, \sigma^2)$ indipendenti, $i = 1, \ldots, 48$. Il confronto si ottiene come mostrato nel seguito.

```
> gatti.lm.rid <- lm(pcuore ~ pcorpo)
```

Il test F per saggiare H_0 contro H_1 è quindi dato da

```
> anova(gatti.lm.rid, gatti.lm)

Analysis of Variance Table

Model 1: pcuore ~ pcorpo
Model 2: pcuore ~ pcorpo + sesso + pcorpo:sesso
  Res.Df    RSS Df Sum of Sq      F Pr(>F)
1     46 76.778
2     44 72.073  2     4.705 1.4362 0.2488
```

La differenza tra i due modelli non è significativa,e si conclude con l'accettazione dell'ipotesi nulla. Quindi, tenendo conto del peso del corpo, non esiste una differenza significativa nel peso del cuore tra i gatti di diverso sesso.

Esercizio. Si effettui l'analisi dei residui. ◇

11.2 Analisi dei dati INSULATE.DAT

I dati nel *file* `insulate.dat` sono relativi ad un esperimento di valutazione dell'effetto dell'isolamento termico sul consumo di gas per riscaldamento. Per 26 settimane prima e per 30 settimane dopo la coibentazione termica di un edificio dotato di un impiato di riscaldamento regolato a 20 gradi Celsius, sono state registrate la temperatura media esterna (misurata in gradi Celsius) e il consumo settimanale di gas (in migliaia di piedi cubi).
Si vuole esplorare

- la relazione tra temperature esterna e consumo di gas;
- l'eventuale variazione della relazione prima e dopo la coibentazione.

Si acquisiscano i dati.

```
> Iso <- read.table("insulate.dat", col.names = c("quando",
+     "temp", "cons"))
> attach(Iso)
```

Il primo quesito chiede di esplorare la relazione esistente tra temperatura e il consumo. Per avere un'idea della relazione tra le due variabili, si consideri il diagramma di dispersione.

```
> plot(cons ~ temp)
```

Dalla Figura 11.6 si nota una evidente relazione decrescente tra temperatura e consumo: all'aumentare della temperatura cala il consumo. Si può provare a vedere se questa relazione è diversa prima e dopo l'isolamento termico (come richiesto dal secondo quesito).

```
> plot(cons ~ temp, type = "n")
> points(temp[quando == "prima"], cons[quando ==
+     "prima"])
> points(temp[quando == "dopo"], cons[quando ==
+     "dopo"], pch = 4, col = 2)
```

Si nota chiaramente (cfr, Figura 11.7) come la relazione rimanga decrescente, ma come i livelli di consumo si abbassino, a parità di temperatura, dopo la coibentazione. Questo parrebbe suggerire che, per spiegare il legame tra temperatura e consumo prima e dopo la coibentazione, sia necessario un modello che preveda due rette di regressione con diversa intercetta. Non è chiaro se le due rette debbano avere anche diverso coefficiente angolare.
Si procede allora con l'analisi della covarianza.

```
> iso.lm <- lm(cons ~ temp + quando + temp:quando)
> summary(iso.lm)
```

```
Call:
lm(formula = cons ~ temp + quando + temp:quando)

Residuals:
     Min       1Q   Median       3Q      Max
-0.97802 -0.18011  0.03757  0.20930  0.63803

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)       4.72385    0.11810  40.000  < 2e-16 ***
temp             -0.27793    0.02292 -12.124  < 2e-16 ***
quandoprima       2.12998    0.18009  11.827 2.32e-16 ***
temp:quandoprima -0.11530    0.03211  -3.591  0.00073 ***
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

Residual standard error: 0.323 on 52 degrees of freedom
Multiple R-Squared: 0.9277,        Adjusted R-squared: 0.9235
F-statistic: 222.3 on 3 and 52 DF,  p-value: < 2.2e-16
```

Tutti i coefficienti risultano significativi. Il test F, inoltre, conferma la validità del modello. Pertanto, non solo il livello medio di consumo cambia prima e dopo la coibentazione, ma cambia anche la forza del legame tra consumo e temperatura. Per evidenziare tali relazioni, è possibile aggiungere al diagramma di dispersione precedentemente creato le rette stimate per il consumo prima della coibentazione ($\widehat{\texttt{cons}} = 4.7238\text{-}0.2779\times\texttt{temp}$) e dopo la coibentazione ($\widehat{\texttt{cons}} = 6.8538\text{-}0.3932\times\texttt{temp}$).

```
> abline(coef(iso.lm)[1:2], col = 2)
> abline(coef(iso.lm)[1:2] + coef(iso.lm)[3:4])
```

Esercizio. Si esegua e commenti l'analisi dei residui del modello. ◇

In questo caso, la stima delle due rette di regressione ottenuta mediante l'analisi della covarianza coincide con la stima di due rette di regressione semplice per i due gruppi di dati separatamente, come si può verificare con le analisi seguenti.

```
> iso.lm.prima <- lm(cons ~ temp, subset = (quando ==
+     "prima"))
> summary(iso.lm.prima)

Call:
lm(formula = cons ~ temp, subset = (quando == "prima"))

Residuals:
     Min       1Q   Median       3Q      Max
```

```
-0.62020 -0.19947  0.06068  0.16770  0.59778

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  6.85383    0.11842   57.88   <2e-16 ***
temp        -0.39324    0.01959  -20.08   <2e-16 ***
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

Residual standard error: 0.2813 on 24 degrees of freedom
Multiple R-Squared: 0.9438,        Adjusted R-squared: 0.9415
F-statistic: 403.1 on 1 and 24 DF,  p-value: < 2.2e-16

> iso.lm.dopo <- lm(cons ~ temp, subset = (quando ==
+     "dopo"))
> summary(iso.lm.dopo)

Call:
lm(formula = cons ~ temp, subset = (quando == "dopo"))

Residuals:
     Min       1Q   Median       3Q      Max
-0.97802 -0.11082  0.02672  0.25294  0.63803

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  4.72385    0.12974   36.41  < 2e-16 ***
temp        -0.27793    0.02518  -11.04 1.05e-11 ***
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

Residual standard error: 0.3548 on 28 degrees of freedom
Multiple R-Squared: 0.8131,        Adjusted R-squared: 0.8064
F-statistic: 121.8 on 1 and 28 DF,  p-value: 1.046e-11
```

Le due rette stimate sono effettivamente identiche. Tuttavia, l'analisi della covarianza offre il vantaggio di poter condurre test sull'uguaglianza dei coefficienti nei due modelli di regressione, oltre a utilizzare una numerosità campionaria più elevata per la stima della varianza.

Per riottenere il grafico in Figura 11.8 con le due rette stimate si puùo anche procedere nel modo seguente.

```
> plot(cons ~ temp, type = "n")
> points(temp[quando == "prima"], cons[quando ==
+     "prima"])
> points(temp[quando == "dopo"], cons[quando ==
+     "dopo"], pch = 4, col = 2)
```

```
> abline(iso.lm.prima$coeff)
> abline(iso.lm.dopo$coeff, col = 2)
```

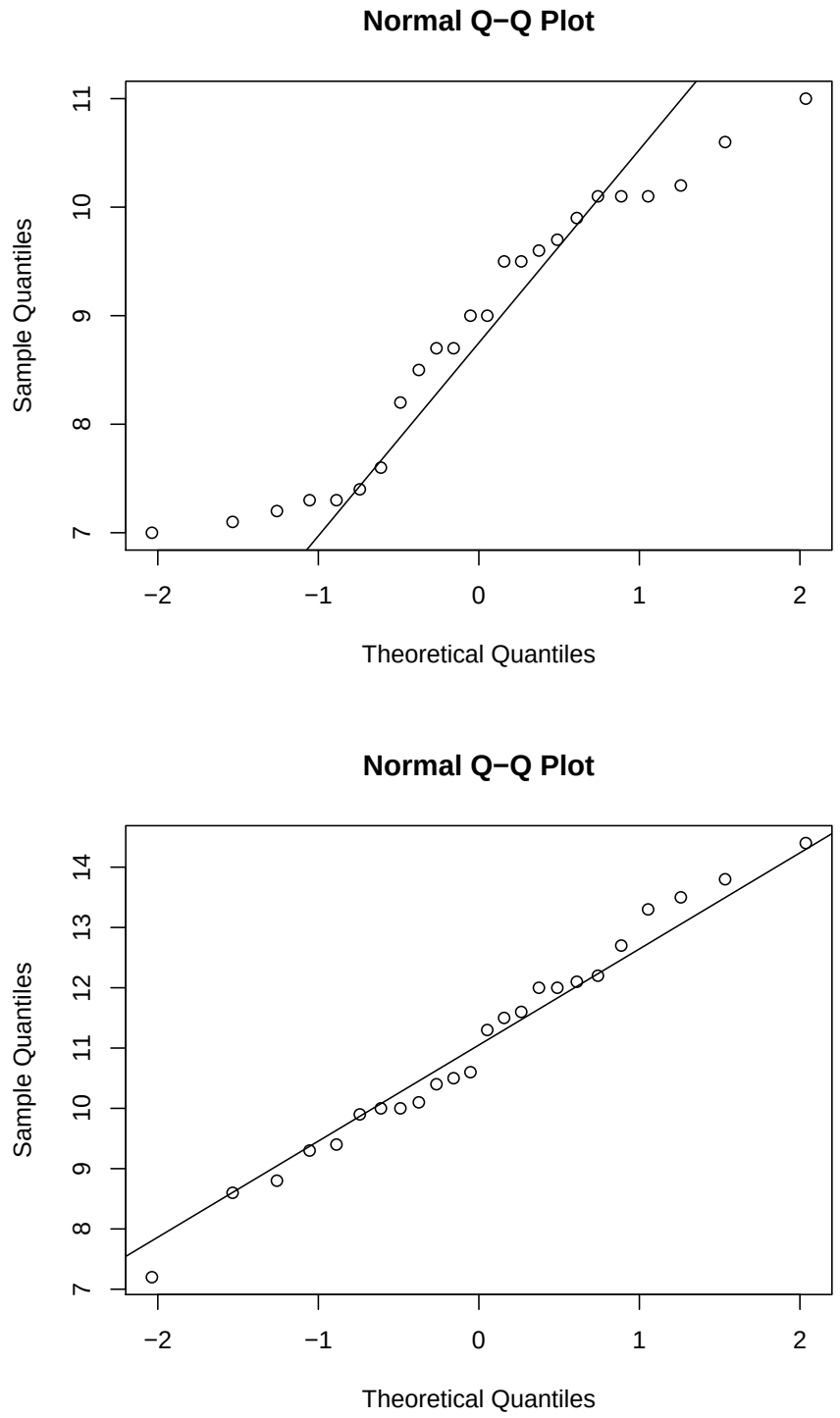


Figura 11.2: *qq-plot* per il peso del cuore nelle femmine (in alto) e nei maschi (in basso).

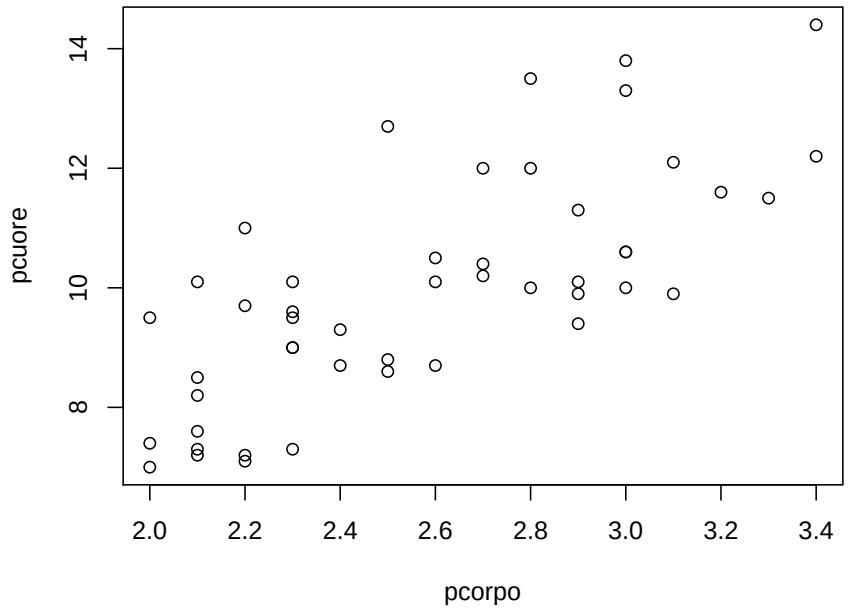


Figura 11.3: Diagramma di dispersione del peso del cuore rispetto al peso del corpo.

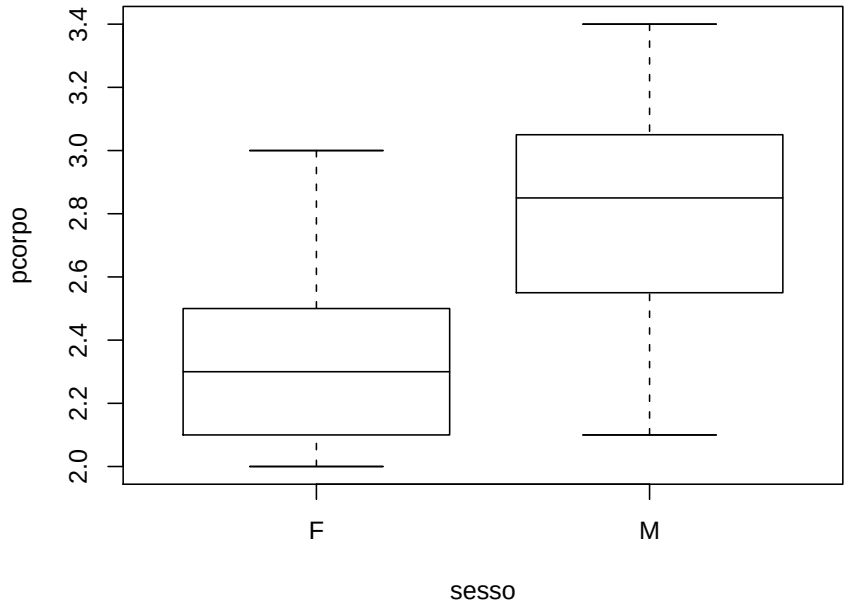


Figura 11.4: Diagrammi a scatola del peso del corpo per i due sessi.

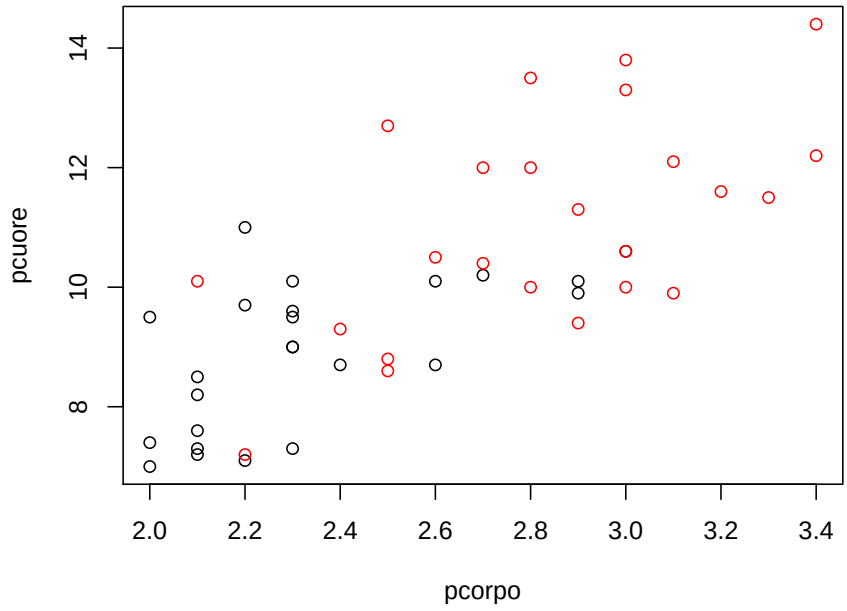


Figura 11.5: Diagramma di dispersione del peso del cuore rispetto al peso del corpo distinguendo i due sessi.

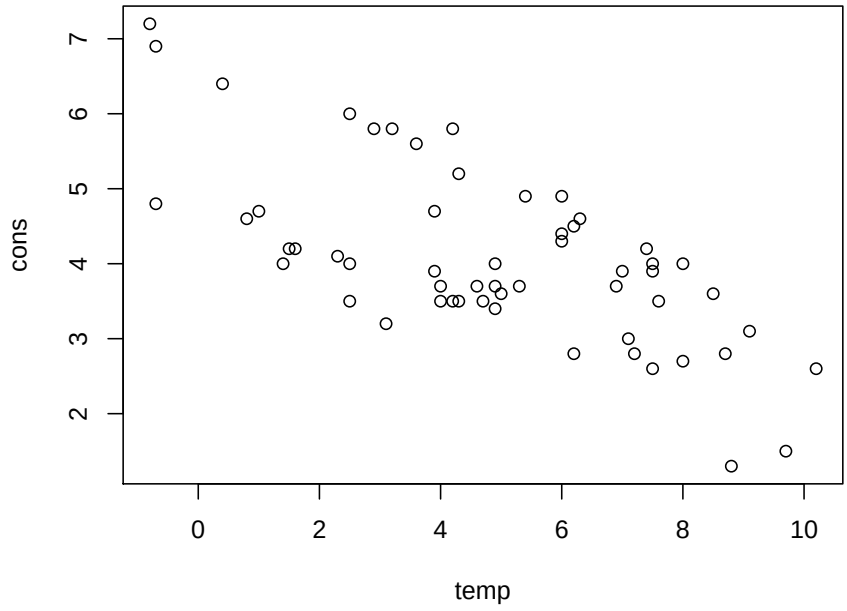


Figura 11.6: Diagramma di dispersione del consumo rispetto alla temperatura.

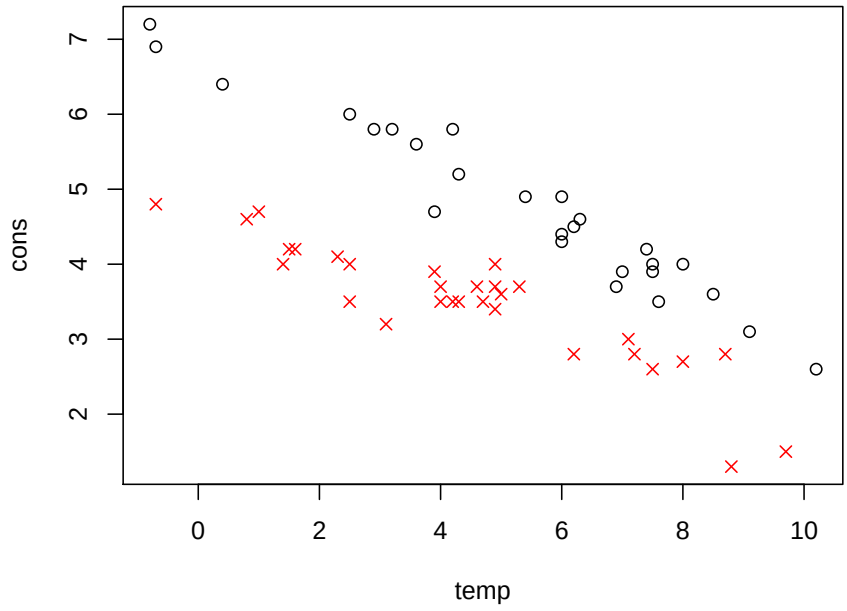


Figura 11.7: Diagramma di dispersione del consumo rispetto alla temperatura, distinguendo rispetto alla variabile `quando`.

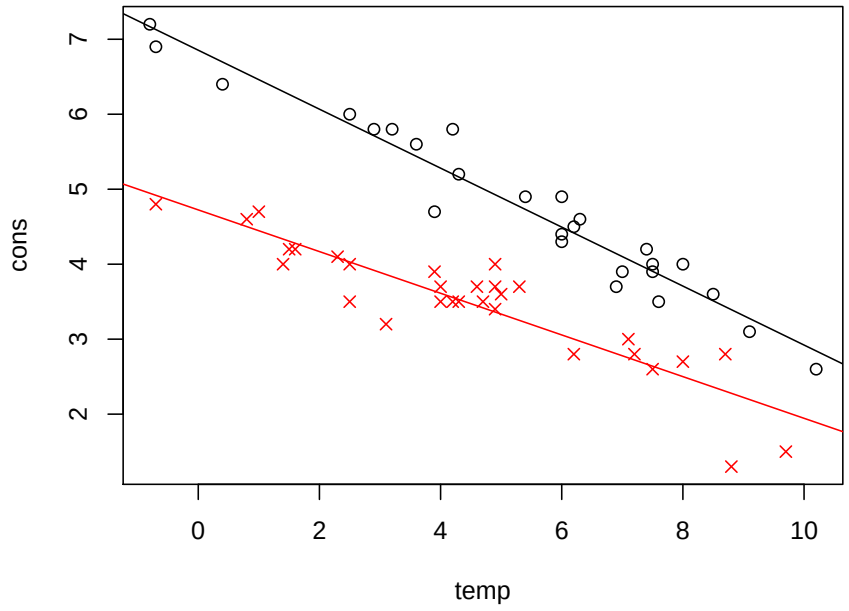


Figura 11.8: Diagramma di dispersione del consumo rispetto alla temperatura, distinguendo rispetto alla variabile `quando`, con le rette di regressione stimate.