Elaborazione Statistica e Numerica dei Dati Sperimentali

G. Graziani (INFN, Sezione di Firenze)

Dispense del corso a.a. 2007/2008

Edizione aggiornata 2014

Indice

Introduzione

Questo corso intende introdurre lo studente, attraverso esempi pratici, ai metodi di analisi statistica dei dati ottenuti in esperimenti scientifici così come da ogni altro tipo di rilevazione.

Lo scopo è quello di mostrare come si possa estrarre la maggior informazione possibile da un insieme di dati, tenendo conto delle incertezze casuali e sistematiche che inevitabilmente limitano la precisione e l'affidabilità di qualunque osservazione sperimentale.

Questo necessita innanzi tutto di comprendere, tramite la teoria della probabilità, l'effetto delle incertezze sui dati sperimentali.
Si dovrà poi invertire il problema, per stimare il valore delle variabili teoriche di interesse e la relativa incertezza a partire da un campione di dati, utilizzando le tecniche di statistica inferenziale.

I concetti base di teoria della probabilità sono richiamati nel cap.1.
I capitoli successivi trattano vari problemi di statistica inferenziale attraverso esempi ed esercizi, dopo aver introdotto i relativi principi matematici. Il capitolo 2 introduce alle tecniche di visualizzazione dei dati (statistica descrittiva) e alla loro analisi quantitativa tramite stimatori. Sono inoltre brevemente trattate tecniche analitiche e numeriche di simulazione di campioni di dati.
Nel capitolo 3 sono illustrati alcuni dei più comuni tests statistici di ipotesi che permettono una valutazione quantitativa dell'accordo fra un campione di dati ed una data ipotesi teorica.
I modelli parametrici sono trattati nel capitolo 4, a partire dalla semplice analisi di regressione lineare, fino alla risoluzione numerica di problemi non lineari.
Viene infine affrontato, nel capitolo 5, il problema delle incertezze sistematiche nelle misure .

Gli esempi di analisi dei dati sono svolti con l'ausilio dell'ambiente software open–source R (http://www.r-project.org), ampiamente documentato sulla sua pagina web (si consiglia in particolare il manuale [12])

Capitolo 1

Richiami di Teoria della Probabilità

Per una trattazione più esaustiva e rigorosa, si faccia riferimento al corso di Probabilità e Statistica, e ai testi contenuti in Bibliografia, in particolare [3], [1].

1.1 Misure sperimentali ed incertezza

Nell'ambito della fisica classica, che tratta modelli deterministici, il risultato di una misura sperimentale dovrebbe idealmente essere univocamente determinato dalla conoscenza dello stato del sistema considerato.

Anche senza spingersi a considerare gli effetti di fisica quantistica, che in ultima analisi introducono una componente intrinseca di casualità nel processo di misura, il risultato di ogni esperimento pratico è soggetto a variazioni casuali non predicibili a priori, che derivano dall'impossibilità di conoscere tutte le variabili fisiche del sistema con infinita precisione.

Se ad esempio vogliamo conoscere la massa m di un oggetto pesandolo su una bilancia, il risultato della misurazione m_{exp} dipenderà, oltre che da m, da effetti difficilmente controllabili (attriti negli ingranaggi della bilancia, polvere, flussi d'aria, etc).
Si avrà dunque un **errore** nella misurazione pari a

$$\epsilon = m_{exp} - m$$

che varia in generale in modo casuale di misura in misura. Queste fluttuazioni, che non portano informazioni su m, limitano la precisione della misura introducendo un'incertezza nella determinazione di m [1].
La variabile m_{exp} è un esempio di **variabile aleatoria**, ovvero di una variabile che anzichè

[1] l'errore della misura ha in generale anche una componente sistematica, uguale per ogni misurazione, dovuta ad esempio una errata calibrazione della bilancia. Questo tipo di incertezza, che qui per ora non consideriamo, sarà discussa nel capitolo 5

avere un unico valore ben definito (come m), può assumere un valore diverso per ciascuna osservazione in modo non prevedibile.

Il valore di m_{exp} non può dunque essere predetto, anche ammettendo di conoscere a priori il valore esatto di m. Tuttavia, assumendo che gli effetti che determinano l'incertezza non dipendano dal tempo, ovvero che la misura sia **riproducibile**, possiamo comunque aspettarci che le fluttuazioni aleatorie seguano una legge matematica. Dato un certo intervallo di valori $A = [m_a, m_b]$, ci aspettiamo di poter predire, se non il valore di ogni singola misura, almeno la frequenza $N(A)/N$ con cui il risultato $m_{exp} \in A$ si manifesta dopo un numero N di misure. Il numero finito di osservazioni limita i possibili valori della frequenza, ma è ragionevole assumere che il limite per infinite misure

$$P(A) = lim_{N\to\infty} \frac{N(A)}{N} \tag{1.1}$$

abbia un valore ben definito che chiameremo **probabilità** di ottenere il risultato A.
La relazione 1.1 non va intesa come una definizione, essendo una relazione empirica non verificabile in pratica. Per procedere è dunque necessario introdurre una definizione di probabilità più formale.

1.2 Teoria assiomatica di Kolmogorov

La più comune formulazione di teoria della probabilità segue l'approccio assiomatico di Kolmogorov[8]. La probabilità P è definita come un numero reale che ha un valore definito per ogni elemento A di un insieme S, detto "spazio degli eventi", tale da soddisfare i seguenti tre assiomi:

- $P(A) \geq 0$
- $A \cap B = \emptyset \implies P(A \cup B) = P(A) + P(B)$
- $P(S) = 1$

da cui si possono facilmente ricavare le proprietà che ci aspettiamo da una definizione consistente di probabilità:
$0 \leq P(A) \leq 1$
$P(\emptyset) = 0$
$P(\bar{A}) = 1 - P(A)$
$P(A \cup B) = P(A) + P(B) - P(A \cap B)$

1.3 Probabilità condizionata e teorema di Bayes

Si definisce la **probabilità condizionata** di A dato B

$$P(A|B) \equiv \frac{P(A \cap B)}{P(B)} \tag{1.2}$$

Gli elementi A e B si dicono **indipendenti** se e solo se $P(A \cap B) = P(A)P(B)$

In tal caso la probabilità condizionata $P(A|B) = P(A)$ è indipendente da B.

Dalla definizione 1.2 e dalla relazione $A \cap B = B \cap A$ consegue il **teorema di Bayes**

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{1.3}$$

Il teorema è molto utile in pratica, in particolare se possiamo suddividere l'insieme S in sottoinsiemi disgiunti A_i:

$S = \bigcup_i A_i$

$A_i \cap A_j = \emptyset \quad i \neq j$

nel qual caso

$P(B) = P(B \cap S) = P(B \cap (\bigcup_i A_i)) = P(\bigcup_i (B \cap A_i)) =$

$= \sum_i P(B \cap A_i) = \sum_i P(B|A_i)P(A_i)$

e il teorema di Bayes può essere riscritto nella forma

$$P(A_x|B) = \frac{P(B|A_x)P(A_x)}{\sum_i P(B|A_i)P(A_i)} \tag{1.4}$$

1.4 Interpretazione frequentista della probabilità

L'approccio assiomatico permette di dare una definizione astratta, matematicamente soddisfacente, di probabilità. Possiamo ora interprere la teoria nei termini del nostro problema di partenza, ovvero la descrizione quantitativa delle incertezze sperimentali.

L'interpretazione "classica", o frequentista, della probabilità consiste nel pensare gli elementi dello spazio S come tutti i possibili valori delle variabili aleatorie che rappresentano i risultati di una misura. L'unione di due elementi costituisce l'OR logico dei due risultati (almeno uno dei due risultati si è verificato), mentre l'intersezione costituisce l'AND logico (entrambi i risultati si sono verificati).

La relazione 1.1 diventa un'**interpretazione**, più che una definizione, di probabilità, che è consistente con la teoria poiché possiamo facilmente verificare che rispetta i tre assiomi di Kolmogorov.

Illustriamo ora l'importanza pratica del teorema di Bayes tramite un esempio: supponiamo di voler diagnosticare una certa malattia attraverso un test clinico su un campione, arbitrariamente grande, di individui. Vi sono due variabili aleatorie in gioco:

- ogni individuo può essere sano (S) o malato (M)
- il test può dare un risultato positivo ($+$) o negativo ($-$)

I costituenti elementari dell'insieme S sono dunque 4: ($M+$), ($M-$), ($S+$), ($S-$).
Il teorema di Bayes ci permette di ottenere l'informazione che ci interessa (la probabilità di essere sano o malato) a partire dal dato accessibile sperimentalmente (il risultato del test). Per questo è necessario conoscere le probabilità, detta "a priori", di contrarre la malattia:
$P(M) = 0.1\%$
e l'affidabilità del test, espressa dalle probabilità condizionate
$P(+|M) = 99.6\%$
$P(+|S) = 1.7\%$
La probabilità "a posteriori" di essere malati in presenza di un risultato positivo del test sarà

$$P(M|+) = \frac{P(+|M)P(M)}{P(+|M)P(M) + P(+|S)P(S)} = 5.5\%$$

Supponiamo ora che la probabilità a priori $P(M) = 0.1\%$ sia riferita alla popolazione italiana, e di sapere che la probabilità è più elevata per la popolazione di Firenze:
$P(M|Fi) = 7.5\%$
In tal caso, la probabilità di essere malato per gli abitanti di Firenze che risultano positivi al test sarà

$$P(M|+, Fi) = \frac{P(+|M)P(M|Fi)}{P(+|M)P(M|Fi) + P(+|S)P(S|Fi)} = 82.6\%$$

Supponiamo ora che il signor Tizio, fiorentino, si sottoponga al test e risulti positivo. Potrebbe domandarsi: qual è la probabilità che io sia malato? L'82.6 % ? Ma il signor Tizio fa anche parte della popolazione italiana, per cui abbiamo ricavato un valore molto più basso (5.5%).
Rigorosamente, nell'approccio frequentista questa domanda è semplicemente mal posta: la variabile "il signor Tizio è malato" non è aleatoria, poiché ha necessariamente un valore definito (vero o falso), anche se lo ignoriamo, e la sua probabilità non è definibile.

1.5 Interpretazione bayesiana (soggettiva) della probabilità

L'approccio frequentista, pur essendo rigoroso ed oggettivo, non risulta soddisfacente in molti problemi pratici. Ad esempio la domanda del signor Tizio è tutto sommato ragionevole: dopo-

tutto si è sottoposto al test per sapere se ha contratto la malattia, non per partecipare ad una rilevazione statistica!

Un approccio alternativo al concetto di probabilità è quello Bayesiano (o "soggettivo"), che consiste in una diversa interpretazione della teoria assiomatica della probabilità:

- lo spazio S costituisce un insieme di ipotesi che possono essere vere e false, tali che la loro unione (ovvero l'OR logico di tutte le ipotesi) sia necessariamente vera;

- la probabilità costituisce un "grado di confidenza" che l'ipotesi sia vera, stabilito soggettivamente dall'osservatore.

In questo approccio si rinuncia dunque a una definizione oggettiva di probabilità, che viene definita a priori in modo arbitrario cercando, per usare le parole di Laplace, di "trasformare il buon senso in calcoli". In questo approccio è lecito parlare della probabilità $P(teoria)$ che una certa ipotesi teorica sia vera, e i dati sperimentali possono migliorare la nostra confidenza sull'ipotesi teorica per mezzo del teorema di Bayes:

$$P(teoria|esperimento) = \frac{P(esperimento|teoria)P(teoria)}{P(esperimento)} \tag{1.5}$$

Il problema dell'interpretazione bayesiana è che il risultato dipende dall'assunzione a priori $P(teoria)$ che non è definibile oggettivamente, e infatti non ha alcun senso nell'approccio classico.

La statistica bayesiana cerca di definire delle procedure che, sulla base di un criterio di ragionevolezza, cercano di minimizzare l'effetto delle scelte soggettive sul risultato finale dell'analisi statistica. In particolare, laddove esistano dei dati sperimentali sulla frequenza di un certo risultato, questa sarà usata come miglior stima della probabilità a priori. In questo senso si può dire che l'approccio bayesiano include quello frequentista.
Nel caso del nostro esempio, il signor Tizio stimerà la probabilità a priori di essere malato in base alle informazioni di cui dispone e dunque userà senz'altro il valore $P(M|Fi)$ che è più specifico dell'area in cui vive. Ma se non fosse a conoscenza del dato $P(M|Fi)$ ma solo di quello nazionale, il suo grado di confidenza di avere la malattia cambierebbe drasticamente.

Nel proseguo del corso tratteremo per lo più casi che possono essere trattati col rigore dell'approccio frequentista, accennando al caso bayesiano dove ritenuto opportuno.

1.6 Distribuzioni di probabilità per una variabile aleatoria e loro proprietà

Consideriamo il caso in cui il risultato del nostro esperimento sia rappresentato da una variabile aleatoria a valori reali X.

Si definisce **funzione di ripartizione (o di probabilità cumulativa)** la funzione

$$p(x) = P(X < x) \tag{1.6}$$

che è dunque una funzione monotona crescente con valori fra 0 e 1.
Se questa è derivabile, si definisce **funzione di densità di probabilità (o PDF, da probability density function)**

$$d(x) = \frac{dp(x)}{dx} \tag{1.7}$$

Dalla definizione seguono le proprietâ

$$\int_a^b d(x) = p(b) - p(a) = P(a < X < b) \tag{1.8}$$

$$\int_{-\infty}^{+\infty} d(x) = 1 \tag{1.9}$$

Se $p(x)$ è invertibile, si definisce **funzione quantile**

$$q(x) = p^{-1}(x) \tag{1.10}$$

per $0 \leq x \leq 1$.

La legge di probabilità è espressa indifferentemente da una delle tre funzioni $p(x)$, $d(x)$, $q(x)$, anche se il grafico della PDF $d(x)$ è il modo più comune di rappresentare la distribuzione dei valori.
Il **valore atteso** (o valor medio) di una distribuzione è

$$E(X) = \int_{-\infty}^{+\infty} x d(x) dx \tag{1.11}$$

Analogamente si definisce valore atteso di una generica funzione $f(x)$

$$E(f(X)) = \int_{-\infty}^{+\infty} f(x) d(x) dx \tag{1.12}$$

Si dimostra che la conoscenza di $d(x)$ è equivalente alla conoscenza di $E(X)$ e di tutti i **momenti** della distribuzione

$$\mu_l = E\left((X - E(X))^l\right) \tag{1.13}$$

con $l > 1$ (si ha ovviamente $\mu_0 = 1, \mu_1 = 0$). I momenti quantificano le proprietà essenziali della distribuzione con importanza decrescente al crescere di l. Il momento secondo μ_2 è detto **varianza**

$$\sigma^2 \equiv \mu_2 = \int_{-\infty}^{+\infty} (x - E(X))^2 d(x) dx = E(X^2) - (E(X))^2 \tag{1.14}$$

e la sua radice σ è detta **deviazione standard**.

Nell'analisi dei dati, il valore atteso e la deviazione standard sono i due parametri fondamentali di una distribuzione. Infatti le incertezze in una misura sono tipicamente "a media nulla", ovvero seguono una distribuzione di probabilità con valore atteso nullo. In tal caso, il valore atteso di X coincide con il valore teorico ("valore vero") che vogliamo misurare. La deviazione standard indica invece di quanto i valori misurati si possano discostare dal valore vero ed è il modo convenzionale di quantificare l'**errore** sulla misura.

Il momento terzo è talvolta utilizzato per valutare l'asimmetria della distribuzione tramite il parametro adimensionale $y = \mu_3/\sigma^3$, detto asimmetria. Infatti

$$d(E(X) + \delta) = d(E(X) - \delta) \quad \forall \delta \quad \Longrightarrow \quad y = 0$$

(si noti che non è necessariamente vero il viceversa).
Il momento quarto è invece usato per quantificare quanto una distribuzione è "piccata", tramite il parametro detto curtosi[2]

$$k = \frac{\mu_4}{\sigma^4} - 3$$

Altri parametri spesso utilizzati per descrivere in sintesi una distribuzione di probabilità sono alcuni valori particolari della funzione quantile:
la **mediana** q(1/2)
il primo e terzo **quartile** q(1/4) e q(3/4).
Se la funzione è simmetrica, la mediana coincide evidentemente col valore atteso, e si ha inoltre q(3/4)-q(1/2)=q(1/2)-q(1/4)
Un'altra grandezza indicativa è il valore più probabile, o **moda**, per il quale $d(x)$ ha il valore massimo.

1.6.1 Sul significato della deviazione standard

Come già detto, la varianza è il parametro che misura di quanto una variabile aleatoria possa fluttuare intorno al suo valore atteso. Se tali fluttuazioni sono dovute ad errori nella misura,

[2]la curtosi è definita in modo da valere 0 per la distribuzione normale. La definizione permette inoltre una semplice relazione per la curtosi della somma di più variabili aleatorie:
$k(\sum_1^n X_i) = \sum(k(X_i))/n^2$

può dunque essere usata per quantificare l'incertezza nel processo di misura. La deviazione standard, che ha le stesse dimensioni della variabile X, è usata convenzionalmente per indicare l'"errore" sulla misura. L'interpretazione in termini probabilistici di questo errore, ovvero la probabilità che il valore di X si discosti dal suo valore atteso di non oltre un certo numero di deviazioni standard, dipende dalla distribuzione in esame. Possiamo però stabilire un limite superiore a questa probabilità grazie alla **diseguaglianza di Chebyshev**:

$$\begin{aligned}\sigma^2(X) &= \int_{-\infty}^{+\infty} (x - E(x))^2 \mathcal{d}(x) dx \geq \int_{|x-E(x)|>t} (x - E(x))^2 \mathcal{d}(x) dx \\ &\geq t^2 \int_{|x-E(x)|>t} \mathcal{d}(x) dx = t^2 P(|x - E(x)| > t)\end{aligned} \tag{1.15}$$

per una qualunque costante t. Se prendiamo $t = n\sigma(X)$ otteniamo che la probabilità che il valore di X si discosti di oltre n deviazioni standard dal suo valore atteso è inferiore a n^{-2}

$$P\left(|x - E(x)| > n\sigma(X)\right) \leq 1/n^2 \tag{1.16}$$

indipendentemente dalla distribuzione $\mathcal{d}(x)$!

Ad esempio, la probabilità che il valore oscilli di oltre 5 σ rispetto al valore atteso è sempre inferiore al 4%. Questo risultato giustifica in generale l'uso della deviazione standard per indicare un intervallo di valori in cui possiamo essere confidenti, con una certa probabilità, che il valore si trovi, sebbene per le distribuzioni di uso più frequente la probabilità $P\left(|x - E(x)| > n\sigma(X)\right)$ risulti molto inferiori al limite di Chebyshev.

1.7 Alcuni esempi notevoli di distribuzione

1.7.1 Distribuzione uniforme

Se tutti i valori di una variabile X in un intervallo $A = [a, b]$ sono considerati equiprobabili, la relativa PDF è la distribuzione uniforme

$$\mathcal{d}_u(x) = \frac{1}{(b-a)} \chi_A(x) \tag{1.17}$$

dove $\chi_A(x)$ è la funzione indicatrice dell'intervallo

$$\chi_A(x) = \begin{cases} 1 & x \in A \\ 0 & x \notin A \end{cases} \tag{1.18}$$

Si verifica facilmente che

$$\sigma_u = \frac{(b-a)}{\sqrt{12}} \tag{1.19}$$

nel software R, le funzioni cumulativa, densità e quantile della distribuzione uniforme sono chiamate rispettivamente punif(), dunif(), qunif()

1.7.2 Distribuzione esponenziale

Consideriamo un evento che può avvenire con una probabilità nell'unità di tempo $dP/dt = r$, indipendente dal tempo. Se si osservano diversi eventi di questo tipo, che avvengono tutti con lo stesso valore di r e sono indipendenti fra loro, si parla di **eventi poissoniani**. Questo è ad esempio il caso del decadimento di un nucleo radioattivo, ma anche della prossima estrazione di un certo numero al lotto (amesso che il sorteggio non sia truccato!).
Consideriamo adesso il tempo di attesa t del prossimo evento. Indichiamo con $p_s(t)$ la probabilità che l'evento non avvenga entro un tempo t. La variazione di $p_s(t)$ in un intervallo infinitesimo dt sarà

$$dp_s = -p_s(t) r dt$$

integrando fra 0 e t (e usando $p_s(0) = 1$) si ottiene

$$p_s(t) = e^{-rt} \tag{1.20}$$

La probabilità che l'evento avvenga entro un tempo t è dunque

$$p_e(t) = 1 - p_s(t) = 1 - e^{-rt} \tag{1.21}$$

Questa è la funzione di ripartizione della distribuzione di probabilità detta esponenziale, che è definita per $t \geq 0$. La sua PDF è

$$d_e(t) = \frac{1}{\tau} e^{-t/\tau} \tag{1.22}$$

dove il parametro $\tau \equiv 1/r$ è detto "vita media".

Si noti che il tempo 0 a cui comincia l'attesa è arbitrario. Se l'evento non è avvenuto ad un certo tempo t, possiamo sempre azzerare il tempo e chiederci qual è la probabilità che l'evento avvenga entro un certo tempo t' a partire da t, e la legge di probabilità è esattamente la stessa nella variabile t. Infatti l'indipendenza dal tempo di r implica che l'evento avvenga senza "memooria", ovvero senza tener conto di quanto tempo si è atteso in passato (un fatto su cui dovrebbero riflettere i giocatori dei numeri ritardatari al lotto!).
Il valore atteso e la varianza della distribuzione sono

$$E(t_{exp}) = \int_0^{+\infty} \frac{x e^{-x/\tau}}{\tau} dx = \tau \tag{1.23}$$

$$\sigma^2(t_{exp}) = \int_0^{+\infty} \frac{(x-\tau)^2 e^{-x/\tau}}{\tau} dx = \tau^2 \tag{1.24}$$

1.7.3 Distribuzione normale

La distribuzione di probabilità più importante nella scienza statistica è senz'altro la distribuzione normale o **gaussiana**, una distribuzione con due parametri μ e σ la cui PDF è

$$\phi_G(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(\frac{-(x-\mu)^2}{2\sigma^2}\right) \tag{1.25}$$

La funzione cumulativa e la quantile non sono funzioni elementari, ma i loro valori tabulati sono disponibili nei principali software di analisi. *Nel software R, sono ottenibili tramite le funzioni pnorm() e qnorm()*

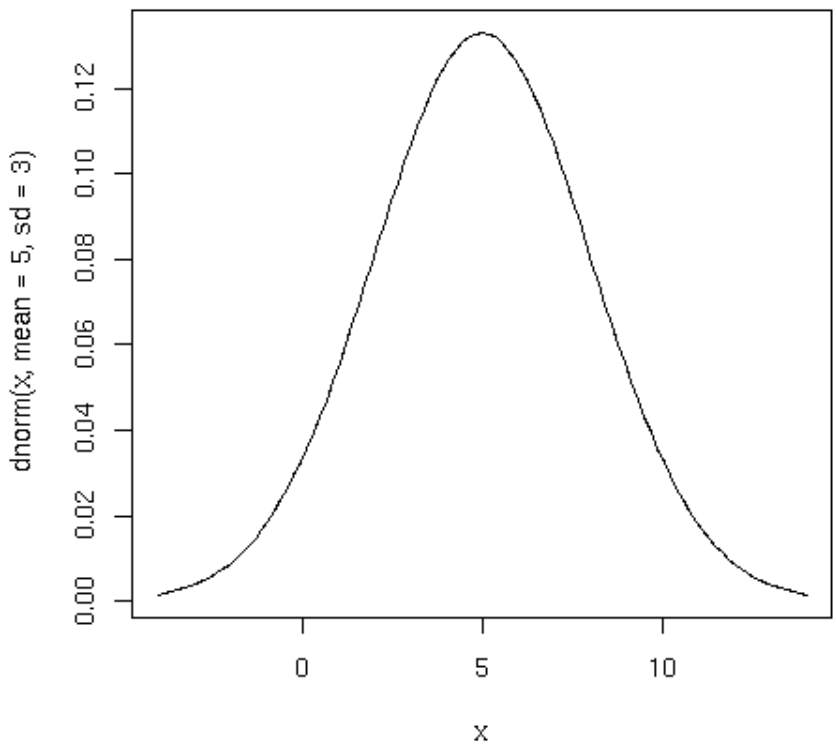


Figura 1.1: Esempio di distribuzione di densità gaussiana. [output del comando *curve(dnorm(x, mean=5, sd=3),-4,14)*]

Essendo una distribuzione simmetrica attorno a μ, il valore atteso è μ. Si dimostra poi che la deviazione standard è pari a σ.

Una notevole proprietà della gaussiana è che la probabilità dipende solo da

$u = (x - \mu)/\sigma$,

ovvero dalla distanza dal valore atteso espressa in "sigma":

$$\phi_{std}(u) = \frac{1}{\sqrt{2\pi}} \exp\left(\frac{-u^2}{2}\right) \tag{1.26}$$

La distribuzione 1.26, che ha valore atteso $E(u) = 0$ e varianza $\sigma^2(u) = 1$, è detta "gaussiana standard".

La funzione di ripartizione della gaussiana standard permette di calcolare la probabilità che la variabile sia pari al valore atteso entro N_σ deviazioni standard, tramite la relazione

$$P(|u| < N_\sigma) = \int_{-N_\sigma}^{N_\sigma} \phi_{std}(u)du = 2p_{std}(N_\sigma) - 1 \tag{1.27}$$

I valori numerici possono essere ottenuti in R dal seguente codice

```
> prob.std = function (Nsigma) { 2*pnorm(Nsigma)-1 }
> prob.std( c(1, 2, 3, 4) )
[1] 0.6826895 0.9544997 0.9973002 0.9999367
```

Una variabile aleatoria gaussiana ha dunque una probabilità del 68.27% di assumere un valore entro 1 σ dal valore atteso, del 95.45% entro 2 σ, del 99.994% entro 3 σ.

L'importanza della distribuzione normale consegue dal teorema del limite centrale, che sarà discusso nel paragrafo 1.13

1.8 Distribuzioni multivariate, covarianza

Le distribuzioni di probabilità definite nel paragrafo precedente possono essere estese al caso di più variabili aleatorie a valori reali $X_i, i = 1 \dots n$. La funzione di ripartizione sarà

$$p(x_1, x_2, \dots x_n) = P\left((X_1 < x_1) \cap (X_2 < x_2) \cdots \cap (X_n < x_n)\right) \tag{1.28}$$

e la funzione densità

$$d(x_1, x_2, \dots x_n) = \frac{\partial^n p}{\partial x_1 \partial x_2 \dots \partial x_n} \tag{1.29}$$

Per ciascuna variabile, si definisce funzione di **probabilità marginale** (o non condizionata)

$$m_{X_i}(x_i) = \int d(x_1, x_2, \dots x_n) \left(\prod_{j \neq i} dx_j \right) \tag{1.30}$$

mentre la definizione di densità di **probabilità condizionata**

$$c_{XY}(y|x) = \frac{d(x, y)}{m_X(x)} \tag{1.31}$$

permette di scrivere il teorema di Bayes per variabili continue

$$c_{XY}(y|x) = \frac{c_{YX}(x|y) m_Y(y)}{m_X(x)} \tag{1.32}$$

Il valore atteso di una funzione $f(X_1 \dots X_n)$ è

$$E(f(X_1 \dots X_n)) = \int f(x_1 \dots x_n) d(x_1, x_2, \dots x_n) dx_1 dx_2 \dots dx_n \tag{1.33}$$

Nel caso di due variabili X e Y, i momenti sono definiti come

$$\mu_{lm} = \int (x - E(X))^l (y - E(Y))^m d(x, y) dx dy \tag{1.34}$$

I momenti μ_{20} e μ_{02} rappresentano le varianze delle due variabili:

$$\sigma^2(X) = \mu_{20} = \int (x - E(X))^2 d(x, y) dx dy = \int (x - E(X))^2 d_m(x) dx \tag{1.35}$$

Il momento μ_{11} è detto **covarianza**

$$\begin{aligned} cov(X, Y) = \mu_{11} &= \int (x - E(X))(y - E(Y)) d(x, y) dx dy \\ &= E(XY) - E(X)E(Y) \end{aligned} \tag{1.36}$$

e risulta particolarmente utile per quantificare la possibile dipendenza fra due variabili.
Due variabili X e Y si dicono indipendenti se e solo se
$P(X \in A, Y \in B) = P(X \in A)P(Y \in B) \quad \forall A, B$
Ne consegue che
X e Y indipendenti $\iff d(x, y) = m_X(x) m_Y(y)$
Per variabili indipendenti la covarianza è nulla:

$$\begin{aligned} & d(x, y) = m_X(x) m_Y(y) \\ & \implies cov(X, Y) = \int (x - E(X))(y - E(Y)) d(x, y) dx dy = \\ & \qquad \int (x - E(X)) m_X(x) dx \int (y - E(Y)) m_Y(y) dy = 0 \end{aligned} \tag{1.37}$$

Una covarianza non nulla mette quindi in evidenza una dipendenza fra le due variabili aleatorie. Si noti che non è vero il viceversa: una covarianza nulla non implica l'indipendenza fra le variabili. Se ad esempio la funzione $m_X(x)$ è simmetrica e il valore di Y è determinato univocamente dal valore di X tramite la relazione $Y = X^2$, la covarianza risulta pari a zero.

Se però la relazione fra Y e X è di tipo lineare, allora il **coefficiente di correlazione lineare**

$$\rho = \frac{cov(X, Y)}{\sigma(X)\sigma(Y)} \tag{1.38}$$

può essere considerato come una misura della mutua dipendenza fra le variabili. Si dimostra facilmente che $-1 \leq \rho \leq 1$. Quando il valore di ρ è negativo si dice che le variabili sono anti–correlate.

La variabilità del nostro set $\underline{x} = \begin{pmatrix} x_1 \\ \dots \\ x_n \end{pmatrix}$ è dunque descritta al primo ordine dai valori delle varianze e delle covarianze. Per i calcoli è utile introdurre la **matrice varianza–covarianza**

$$V = E\left[(\underline{x} - E(\underline{x}))(\underline{x} - E(\underline{x}))^T\right] = \begin{pmatrix} \sigma^2(X_1) & cov(X_1, X_2) & \dots & cov(X_1, X_n) \\ cov(X_2, X_1) & \sigma^2(X_2) & & \\ \dots & & & \\ & & & \sigma^2(X_n) \end{pmatrix} \tag{1.39}$$

La matrice varianza–covarianza delle variabili normalizzate $u_i = (x_i - E(x_i))/\sigma(x_i)$ è detta matrice di correlazione

$$R = \begin{pmatrix} 1 & \rho(X_1, X_2) & \dots & \rho(X_1, X_n) \\ \rho(X_2, X_1) & 1 & & \\ \dots & & & \\ & & & 1 \end{pmatrix} \tag{1.40}$$

1.8.1 Distribuzione normale multivariata

Come esempio di funzione densità a più variabili consideriamo la distribuzione normale gaussiana multivariata:

$$\phi_G(\underline{x}) = \frac{1}{(2\pi)^{n/2}\sqrt{detV}} \exp\left(-\frac{1}{2}(\underline{x} - \underline{\mu})^T V^{-1}(\underline{x} - \underline{\mu})\right) \tag{1.41}$$

dove $\underline{\mu} = E(\underline{x})$ è il vettore dei valori attesi e la matrice simmetrica V risulta coincidere con la matrice varianza–covarianza del set di variabili $\underline{x}$.

Questa distribuzione descrive variabili gaussiane (ciascuna distribuzione marginale è una gaussiana unidimensionale) che presentano correlazioni di tipo lineare:

$$E(x_i) = a + bE(x_j)$$

Nel caso bidimensionale, la forma esplicita della funzione è

$$\phi_G(x, y) = \frac{1}{2\pi\sigma(X)\sigma(Y)\sqrt{1-\rho^2}} \cdot \exp\left\{-\frac{1}{2(1-\rho^2)}\left[\left(\frac{x - E(X)}{\sigma(X)}\right)^2 + \left(\frac{y - E(Y)}{\sigma(Y)}\right)^2 - 2\rho\frac{(x - E(X))}{\sigma(X)}\frac{(y - E(Y))}{\sigma(Y)}\right]\right\} \tag{1.42}$$

Notiamo che i valori che hanno pari densità nel piano (X, Y) definiscono una ellisse

$$\phi_G(x, y) = k \quad \Longrightarrow \quad \left(\frac{x - E(X)}{\sigma(X)}\right)^2 + \left(\frac{y - E(Y)}{\sigma(Y)}\right)^2 - 2\rho\frac{(x - E(X))}{\sigma(X)}\frac{(y - E(Y))}{\sigma(Y)} = k' \tag{1.43}$$

detta ellisse di covarianza (k e k' sono due valori costanti).

Il seguente codice permette di visualizzare la funzione con R, come grafico tridimensionale o disegnando nel piano (X, Y) le ellissi di covarianza per diversi valori di k

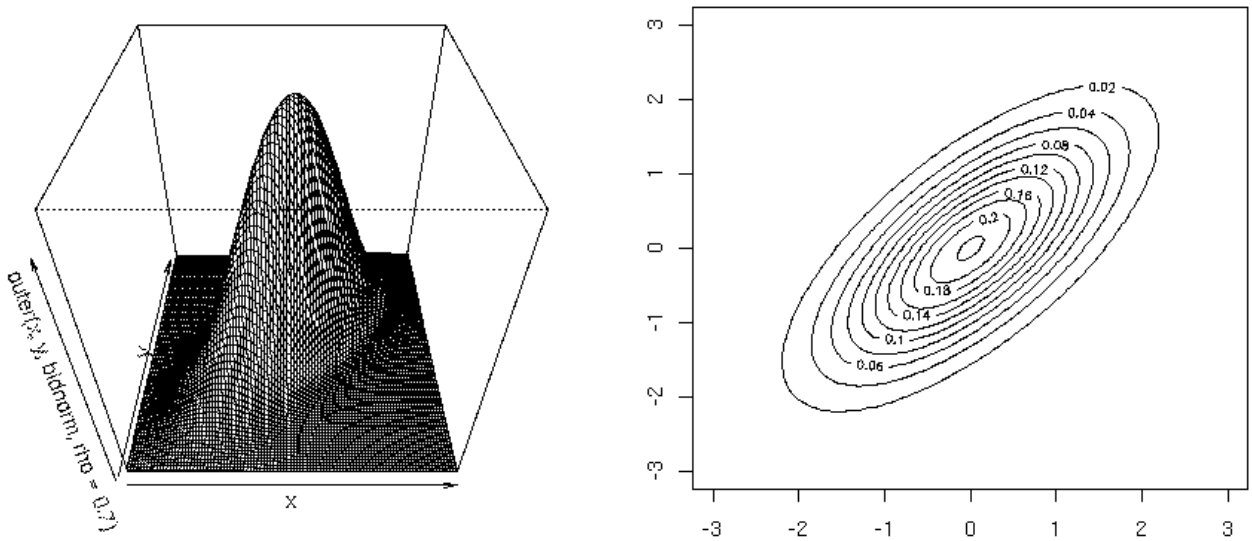


Figura 1.2: Esempio di distribuzione di densità gaussiana bivariata

```
bidnorm = function(x,y,rho=0,sx=1,sy=1 ) {
 1/(2*pi*sx*sy*sqrt(1-rho^2))*
 exp(-1/(2*(1-rho^2))*((x/sx)^2+(y/sy)^2-2*rho*x/sx*y/sy))
}
x=seq(-3,3,length=100)
y=seq(-3,3,length=100)
persp(x,y,outer(x,y,bidnorm,rho=.7),phi=40)
contour(x,y,outer(x,y,bidnorm,rho=.7))
```

Analogamente al caso della gaussiana unidimensionale, possiamo definire delle regioni in cui le fluttuazioni sono "entro $n\sigma$", definite dalle ellissi di covarianza

$$\phi_G(x,y) = \phi_G(E(X), E(Y)) exp\left(-\frac{n^2}{2}\right)$$

La probabilità che i valori di (X,Y) siano contenuti in queste ellissi per $n = 1, 2, 3$ sono rispettivamente il 39.3%, 86.5%, 98.9%.

1.9 Funzioni di variabili aleatorie

La distribuzione densità $d_y(y)$ per una variabile
y=f(x)

è ottenuta dalla distribuzione $d_x(x)$ imponendo la conservazione della probabilità, che equivale, per variabili reali, ad un cambio di variabile nell'integrazione della funzione densità. Se la funzione $f(x)$ è invertibile si deve avere
$|d_y(y)dy| = |d_x(f^{-1}(y))dx|$ da cui

$$d_y(y) = d_x(f^{-1}(y)) \left| \frac{df^{-1}(y)}{dy} \right| \tag{1.44}$$

Alcuni casi semplici sono

$$y = x + a \implies d_y(y) = d_x(y - a) \tag{1.45}$$

$$y = bx \implies d_y(y) = \left| \frac{1}{b} \right| d_x(y/b) \tag{1.46}$$

relazioni che abbiamo già usato per passare dall'eq. 1.25 all'eq. 1.26.
Per una funzione di più variabili aleatorie $\underline{x} = (x_1, x_2 \dots x_n)$
$W = f(\underline{x})$
la funzione densità si ottiene differenziando la funzione di ripartizione calcolata come

$$p_W(w) = P(f(\underline{x}) < w) = \int_{A(w)} d_x(\underline{x}) d\underline{x} \tag{1.47}$$

dove $A(w)$ rappresenta l'insieme dei valori di $\underline{x}$ per cui $f(\underline{x}) < w$.
Nel caso di due variabili indipendenti X_1 e X_2, si ha

$$p_W(w) = P(f(x_1, x_2) < w) = \int_{-\infty}^{+\infty} d_1(x_1) dx_1 \int_{A(f(x_1,x_2)<w)} d_2(x_2) dx_2 \tag{1.48}$$

e la funzione densità

$$d_W(w) = \frac{dp_W(w)}{dw} = d_1 \otimes d_2$$

è detta **convoluzione** delle funzioni d_1 e d_2.
Casi notevoli sono la **convoluzione di Fourier** per il caso $w = x_1 + x_2$

$$\begin{aligned} d_W(w) =& \frac{d}{dw} \left(\int_{-\infty}^{+\infty} d_1(x_1) dx_1 \int_{-\infty}^{w-x1} d_2(x_2) dx_2 \right) = \\ & \int_{-\infty}^{+\infty} d_1(x_1) d_2(w - x1) dx_1 \end{aligned} \tag{1.49}$$

e la **convoluzione di Mellin** per il caso $z = x_1 \cdot x_2$

$$\begin{aligned} d_Z(z) =& \frac{d}{dz} \left(\int_{-\infty}^{+\infty} d_1(x_1) dx_1 \int_{-\infty}^{z/x1} d_2(x_2) dx_2 \right) = \\ & \int_{-\infty}^{+\infty} d_1(x_1) \frac{1}{|x_1|} d_2 \left(\frac{z}{x_1} \right) dx_1 \end{aligned} \tag{1.50}$$

Esempio 1.9.1 *Convoluzione di due variabili distribuite uniformemente*
Si consideri la variabile

$$y = x_1 + x_2$$

con x_1 e x_2 distribuite uniformemente fra a e b.

La funzione densità di probabilità per la variabile y è

$$g(y) = \int_a^b \frac{1}{(b-a)} f_x(y-x)dx = \int_{max(a,y-b)}^{min(b,y-a)} \frac{1}{(b-a)^2} dx = \begin{cases} 0 & y < 2a \\ \frac{y-2a}{(b-a)^2} & 2a < y < a+b \\ \frac{2b-y}{(b-a)^2} & a+b < y < 2b \\ 0 & y > 2b \end{cases}$$

Si ottiene cioè una funzione triangolare, implementata in R dal seguente codice:

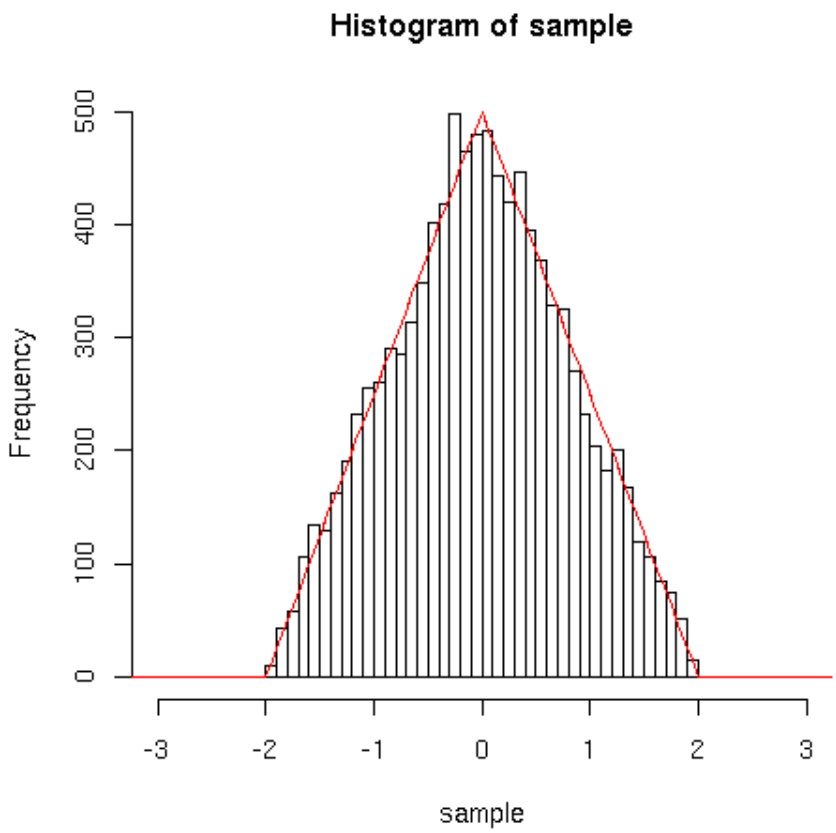
```
hy <-  function(y,a=-1,b=1) {
 (y > 2*a & y < 2*b) *
   ((y-2*a) * (y < (a+b)) + (2*b-y) * (y >= (a+b))) /(b-a)^2
}
```

Possiamo verificare la validità della funzione ottenuta tramite una **simulazione**.
La funzione *runif()* di R ci permette di estrarre a caso dei valori con distribuzione uniforme in un dato intervallo. Possiamo dunque produrre N valori per x_1, x_2 e dunque y. La distribuzione empirica dei valori ottenuti di y può essere rappresentata tramite un **istogramma**, contando la frequenza dei dati in intervalli di y di pari larghezza δ. La funzione di R *hist()* ci permette di produrre e visualizzare l'istogramma; questo può essere confrontato con i valori attesi λ_i in ciascun intervallo $I_i = [y_i - \delta/2, y_i + \delta/2]$

$$\lambda_i = N \cdot P(y \in I_i) = N \cdot \int_{I_i} d_y(y)dy \simeq N \cdot d_y(y_i)\delta$$

Il confronto visivo fra il risultato della simulazione e la distribuzione attesa può dunque essere fatto sovrapponendo all'istogramma la funzione densità, moltiplicata per il fattore $N \cdot \delta$:

```
mya=-1
myb=1
n=10000
x1=runif(n,min=mya,max=myb)
x2=runif(n,min=mya,max=myb)
sample=x1 + x2
deltax=(myb-mya)/20
hist(sample, breaks=seq(2*mya-1,2*myb+1,deltax))
curve(hy(x,a=mya,b=myb) * n * deltax, add=T, col="red")
```



Esempio 1.9.2 *Convoluzione di una funzione densità gaussiana e una esponenziale*

Si supponga di misurare il tempo di decadimento t di un nucleo radioattivo con un apparato dotato di risoluzione temporale σ, ovvero tale che l'errore di misura ϵ possa essere descritto con una gaussiana di valore atteso nullo e deviazione standard σ. Il valore misurato t' è dunque

$$t' = t + \epsilon$$

con le funzioni densità di probabilità di t e ϵ

$$f(t) = \begin{cases} 0 & t < 0 \\ \frac{1}{\tau}\exp(-t/\tau) & t \geq 0 \end{cases}$$

$$g(\epsilon) = \frac{1}{\sqrt{2\pi}\sigma}\exp(-\epsilon^2/\sigma^2)$$

Vogliamo scrivere con R la funzione densità per la variabile t'

$$h(t') = \int_{-\infty}^{\infty} g(\epsilon) f(t' - \epsilon) d\epsilon = \frac{\exp(-t'/\tau)}{\tau}\int_{-\infty}^{t'} g(\epsilon)\frac{\exp(\epsilon/\tau)}{\tau} d\epsilon$$

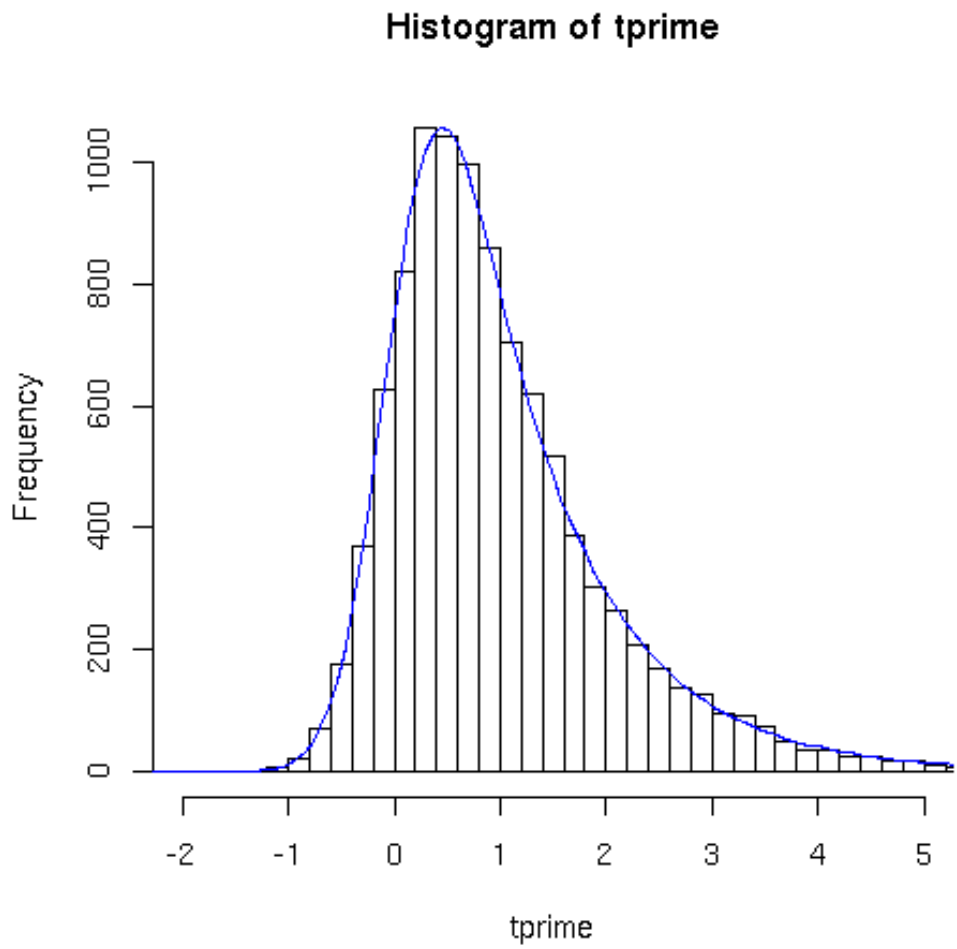
In questo caso la funzione convoluta non è calcolabile analiticamente. Possiamo allora risolvere l'integrale numericamente utilizzando la funzione *integrate()* di R.

```
fee <- function(e,sigma=1,tau=1) {
 dnorm(e,sd=sigma)* exp(e/tau)
}

intfee <- function(t,sigma=1,tau=1) {
  out=c()
  for (i in 1:length(t)) {
    if (i>1) {
        out[i]=out[i-1]
        prevt=t[i-1]
    }
    else {
        out[i]=0.
        prevt=-6.
    }
    out[i] = out[i]+ integrate(fee,prevt,t[i],sigma=sigma,tau=tau)$value
  }
  out
}
expconv <- function(t,sigma=1,tau=1) {
 exp(-t/tau)/tau * intfee(t,sigma,tau)
}
```

(Si noti che la funzione, per poter essere disegnata con le funzioni *plot* o *curve*, deve poter agire su un vettore t di lunghezza arbitraria.)

Come nell'esempio precedente, possiamo simulare un campione e verificare la correttezza della distribuzione densità calcolata. Utilizziamo le funzioni di R per simulare N valori di variabile aleatoria con distribuzione di probabilità esponenziale ($rexp()$) e normale ($rnorm()$)

```
n=10000
mytau=1
mysigma=0.4
t=rexp(n,rate=1/mytau)
eps=rnorm(n,sd=mysigma)
tprime=t+eps
deltax=0.2
hist(tprime,breaks=seq(-10,20,deltax),xlim=c(-2,5))
curve(expconv(x,tau=mytau,sigma=mysigma)*n*deltax,add=T,col="blue")
```



La simulazione di un campione costituisce in effetti una tecnica numerica di calcolo approssimato della distribuzione convoluta, quando questa non sia ricavabile analiticamente. Alcune tecniche di simulazione dei dati saranno brevemente trattate nel paragrafo 2.7.

1.9.1 Propagazione degli errori

Come già sottolineato, il valore atteso e la deviazione standard sono i parametri di maggior interesse nei problemi di analisi dati. Se siamo interessati a conoscere una variabile $y = f(\underline{x})$, nella pratica spesso non è necessario conoscere la forma esatta della distribuzione densità di y, ma è sufficiente saper stimare $E(y)$ e $\sigma(y)$ a partire dai valori misurati di $\underline{x}$ e dai relativi errori, rappresentati dalla matrice varianza–covarianza $V(\underline{x})$. Questo procedimento è chiamato "propagazione degli errori".
Vediamo alcuni casi semplici (a è una costante):

$y = a \cdot x$

$$E(y) = aE(x) \qquad \sigma^2(y) = a^2\sigma^2(x)$$

$y = a + x$

$$E(y) = a + E(x) \qquad \sigma^2(y) = \sigma^2(x)$$

$y = x_1 + x_2$

$$E(y) = \int (x_1 + x_2) d_x(x_1, x_2) dx_1 dx_2 = E(x_1) + E(x_2) \tag{1.51}$$

$$\begin{aligned}
\sigma^2(y) &= E\left(x_1 + x_2 - E(x_1 + x_2)\right)^2 = \\
&= E\left((x_1 - E(x_1)) + (x_2 - E(x_2))\right)^2 = \\
&= E\left((x_1 - E(x_1))\right)^2 + E\left((x_2 - E(x_2))\right)^2 + 2E\left((x_1 - E(x_1)) * (x_2 - E(x_2))\right) = \\
&= \sigma^2(x_1) + \sigma^2(x_2) + 2cov(x_1, x_2)
\end{aligned} \tag{1.52}$$

Se le variabili sono indipendenti, l'errore sulla somma si ottiene semplicemente **sommando in quadratura** gli errori sulle $\underline{x}$.
Per una generica trasformazione lineare

$$y = f(\underline{x}) = a_0 + \sum_{i=1}^{n} a_i x_i$$

si ha dunque

$$E(y) = a_0 + \sum_{i=1}^{n} a_i E(x_i) = f(E(\underline{x})) \tag{1.53}$$

$$\sigma^2(y) = \sum_{i=1}^{n} a_i^2 \sigma^2(x_i) + 2 \sum_{i>j} a_i a_j cov(x_i, x_j) \tag{1.54}$$

Si noti che se le variabili x_i hanno distribuzione normale, tale sarà anche la distribuzione di y, poiché si dimostra che la convoluzione di Fourier di due gaussiane è ancora una gaussiana. In tal caso quindi la distribuzione di y è completamente definita.
Nel caso di una trasformazione lineare da n variabili $\underline{x}$ a m variabili $\underline{y}$:

$$y_j = a_{j0} + \sum_{i=1}^{n} a_{ji} x_i$$

si possono calcolare facilmente le covarianze fra le y_i:

$$cov(y_l, y_k) = \sum_{i=1}^{n} a_{li} a_{ki} \sigma^2(x_i) + 2 \sum_{i>j} a_{li} a_{kj} cov(x_i, x_j) \tag{1.55}$$

Utilizzando la notazione matriciale $\underline{y} = \underline{y}_0 + A\underline{x}$, le equazioni 1.53 e 1.55 possono essere riscritte nella forma più compatta

$$V_y = A V_x A^T \tag{1.56}$$

Veniamo ora al caso non lineare, cominciando col considerare il prodotto e il rapporto di due variabili:

$y = x_1 \cdot x_2$

ricordando la 1.36

$$E(y) = E(x_1)E(x_2) + cov(x_1, x_2) \tag{1.57}$$

dunque il valore atteso del prodotto è il prodotto dei valori attesi solo se le variabili sono indipendenti o comunque scorrelate.

La varianza può essere calcolata nel caso di variabili indipendenti:

$$\begin{aligned} d_x(x_1, x_2) &= d_1(x_1)d_2(x_2) \\ \Longrightarrow \quad \sigma^2(y) &= E(x_1^2)E(x_2^2) - E(x_1)^2E(x_2)^2 = \\ &= \sigma^2(x_1)E(x_2)^2 + \sigma^2(x_2)E(x_1)^2 + \sigma^2(x_1)\sigma^2(x_2) \end{aligned} \tag{1.58}$$

da cui

$$\frac{\sigma(x_1x_2)}{E(x_1x_2)} = \sqrt{\left(\frac{\sigma(x_1)}{E(x_1)}\right)^2 + \left(\frac{\sigma(x_2)}{E(x_2)}\right)^2 + \left(\frac{\sigma(x_1)\sigma(x_2)}{E(x_1)E(x_2)}\right)^2} \tag{1.59}$$

Nel caso di variabili indipendenti con errori relativi $\sigma(x)/E(x) \ll 1$ si può trascurare il terzo termine della somma sotto la radice e si ottiene che l'errore **relativo** del prodotto di variabili aleatorie si ottiene sommando in quadratura gli errori relativi delle $\underline{x}$.

$y = x_1/x_2$

in questo caso

$$E(y) = E(x_1)E(1/x_2) + cov(x_1, 1/x_2) \tag{1.60}$$

ed in generale il valore atteso del rapporto è diverso dal rapporto dei valori attesi. L'approssimazione $E(y) \simeq E(x_1)/E(x_2)$ è valida solo nel caso di errori relativi piccoli sulle x.

Per una generica funzione $y = f(\underline{x})$ non si può dare una formula di propagazione esatta, ma notiamo che sono rilevanti solo i valori della funzione per valori di $\underline{x}$ che hanno una probabilità significativamente maggiore di zero, ovvero che non distano molto, in termini di deviazione standard, dal valore atteso. Ci conviene allora sviluppare la funzione in serie nell'intorno di E($\underline{x}$):

$$f(\underline{x}) = f\left(E(\underline{x})\right) + \sum_i \frac{\partial f}{\partial x_i}\left(E(\underline{x})\right) \cdot \left(x_i - E(x_i)\right) + \ldots \tag{1.61}$$

Se ci limitiamo ad uno sviluppo al primo ordine, approssimando dunque $f(\underline{x})$ ad una funzione lineare nell'intorno di E($\underline{x}$), possiamo utilizzare l'eq. 1.54 per ottenere la nota **formula di propagazione degli errori**

$$\sigma^2(y) \simeq \sum_{i=1}^{n} \left(\frac{\partial f}{\partial x_i}\left(E(\underline{x})\right)\right)^2 \cdot \sigma^2(x_i) + 2\sum_{i>j} \frac{\partial f}{\partial x_i}\left(E(\underline{x})\right) \cdot \frac{\partial f}{\partial x_j}\left(E(\underline{x})\right) \cdot cov(x_i, x_j) \tag{1.62}$$

Bisogna tenere presente che questa formula, molto utilizzata nella pratica, può portare a risultati scorretti quando l'approssimazione lineare non risulti giustificata. Una regola pratica

per sincerarsi dell'applicabilità della formula è la verifica della condizione

$$\left| \frac{[f(E(x)+\sigma(x)) - f(E(x)-\sigma(x)] - \left[\frac{\partial f}{\partial x_i}(E(\underline{x}))\, 2 * \sigma\right]}{[f(E(x)+\sigma(x)) - f(E(x)-\sigma(x))]} \right| \ll 1 \tag{1.63}$$

Esempio 1.9.3 *Errore su una semplice misura*
Si supponga di misurare l'accelerazione gravitazionale g, misurando l'altezza h dopo un tempo t di un grave lasciato cadere (da fermo) da un'altezza nota h_0. I valori osservati di h e t possono essere considerati variabili aleatorie gaussiane con errori $\sigma(h)$ e $\sigma(t)$. Avremo

$$h = h_0 - \frac{g}{2}t^2$$

Nell'approssimazione lineare ci aspettiamo che il valore misurato di g sia una variabile gaussiana con valore atteso

$$E(g) \simeq (h_0 - E(h))\frac{2}{E(t)^2}$$

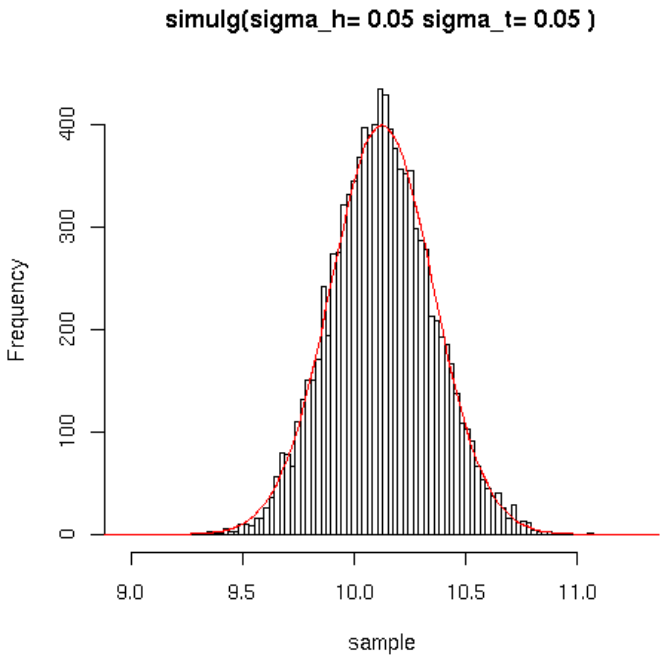
e deviazione standard ottenuta dalla formula di propagazione degli errori:

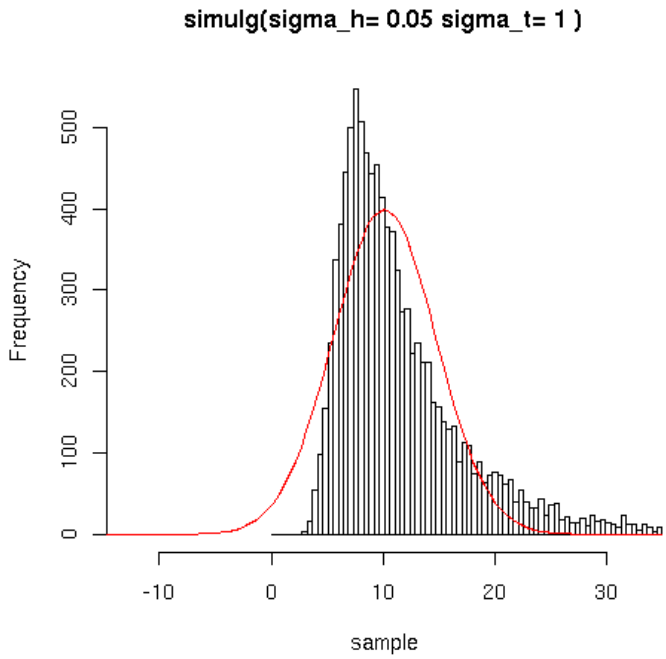
$$\sigma(g) \simeq \sqrt{\frac{4}{E(t)^4}\sigma^2(h) + \frac{16(h_0 - E(h))^2}{E(t)^6}\sigma^2(t)}$$

Verifichiamo la validità dell'approssimazione tramite una simulazione:

```
accelg= function(h0,h,t) {
 (h0-h)*2/t^2
}

simulg = function(sigma_h= 0.05 , sigma_t=0.05) {
 h0=100 # metri
 h=2
 t=4.4  # secondi
 g= accelg(h0,h,t)
 sigma_g = 2/t^2 * sqrt ( sigma_h^2 + 4/t^2*(h0-h)^2*sigma_t^2 )
 cat("g=",g," +/- ",sigma_g," m/s^2\n")

 N=10000
 sample = (h0 - rnorm(N,mean=h,sd=sigma_h))*2 /
       rnorm(N,mean=t,sd=sigma_t)^2

 h=hist(sample,breaks=seq(0,max(sample)+1,sigma_g/10),
        xlim=c(g-5*sigma_g,g+5*sigma_g))
 delta= h$mids[2]-h$mids[1]
 curve(dnorm(x,mean=g,sd=sigma_g)*N*delta,add=T,col="red")
}
```

Possiamo verificare che l'approssimazione lineare è giustificata per i valori di default $\sigma(h) = 0.05$ m e $\sigma(t) = 0.05$ s. Tuttavia, aumentando l'errore su t, la distribuzione di g risulta distorta rispetto all'approssimazione lineare. Si noti che non solo l'errore risulta sottostimato, ma anche il valore più probabile di g si discosta notevolmente da quello atteso nell'approssimazione lineare.





1.10 Variabili aleatorie discrete

Le funzioni di probabilità introdotte nel paragrafo 1.6 possono essere estese al caso di una variabile X a valori discreti $x_1, x_2, \ldots x_N$, sostituendo gli integrali con sommatorie. La funzione discreta densità è semplicemente

$$d(X_i) = P(X_i) \tag{1.64}$$

e la sua condizione di normalizzazione è

$$\sum_{i=1}^{N} d(X_i) = 1 \tag{1.65}$$

Il valore atteso e i momenti sono

$$E(X) = \sum_{i=1}^{N} X_i d(X_i) \tag{1.66}$$

$$\mu_l(X) = \sum_{i=1}^{N} (X_i - E(X))^l \, d(X_i) \tag{1.67}$$

1.11 Distribuzione binomiale e legge dei grandi numeri

La **distribuzione di Bernoulli** è una funzione di variabile discreta X con valori 0 o 1 ("successo") e un parametro p:

$$f_B(X) = \begin{cases} 1-p & X=0 \\ p & X=1 \end{cases} \tag{1.68}$$

e descrive quindi un fenomeno stocastico che può avvenire con probabilità p. Il valore atteso e la varianza sono:

$$E(X) = \sum_{x=0}^{1} X f_B(x) = p \tag{1.69}$$

$$\sigma^2(X) = \sum_{x=0}^{1} (X-p)^2 f_B(x) = p(1-p) \tag{1.70}$$

Se ora consideriamo n variabili aleatorie bernoulliane indipendenti B_i con uguale parametro p, la probabilità di avere un numero $k = \sum B_i$ di successi è data dal prodotto delle probabilità di avere k successi e $n-k$ insuccessi, moltiplicata per il numero di combinazioni di n elementi di lunghezza k, ovvero per il coefficiente binomiale $\binom{n}{k}$:

$$f_b(k; n, p) = \frac{n!}{k!(n-k)!} p^k (1-p)^{n-k} \tag{1.71}$$

La funzione densità ottenuta è detta **distribuzione binomiale** e si applica in particolare al caso di una **misura campionaria** in cui si vuole stimare la probabilità p di un certo evento tramite n osservazioni **indipendenti**, k delle quali manifestano l'evento osservato.
Il valore atteso di k è

$$E(k) = E(\sum_{i=1}^{n} B_i) = n \cdot p \tag{1.72}$$

e la varianza

$$\sigma^2(k) = \sum_{i=1}^{n} \sigma^2(B_i) = n \cdot p \cdot (1-p) \tag{1.73}$$

Per la frequenza $f = k/n$ di successi si ha dunque

$$E(f) = p \tag{1.74}$$

$$\sigma(f) = \sqrt{\frac{p \cdot (1-p)}{n}} \tag{1.75}$$

L'equazione 1.75, che giustifica a posteriori l'interpretazione di probabilità 1.1, esprime un concetto noto come "legge dei grandi numeri": il valore della frequenza f converge alla probabilità p dell'evento nel limite $n \to \infty$. Le fluttuazioni statistiche di f, e dunque l'errore nella determinazione di p, sono tanto più piccole quanto maggiore è n. Come vedremo meglio nel prossimo capitolo, in conseguenza di questa legge l'errore statistico su una generica stima

campionaria è inversamente proporzionale alla radice della dimensione n del campione, e può dunque essere reso arbitrariamente piccolo a patto di disporre di un campione sufficientemente grande.

La distribuzione binomiale può essere estesa al caso in cui la variabile X_B può assumere un qualunque numero di valori discreti X_i , ciascuno con probabilità p_i, tali che $\sum p_i = 1$. In tal caso la distribuzione è detta **multinomiale**:

$$f_{mn}(k_1, k_2 \dots k_{m-1}; n, p_1, p_2 \dots p_{m-1}) = \frac{n!}{\prod_{j=1}^{m} k_j!} \prod_{j=1}^{m} p_j^{k_j} \tag{1.76}$$

Per ciascuna variabile k_i sono ancora valide le formule 1.72 e 1.73, e il calcolo della covarianza fra le variabili conduce a

$$cov(k_i, k_j) = \sum_{k_1=1}^{n} \sum_{k_2=1}^{n} \cdots \sum_{k_m=1}^{n} (k_i - E(k_i))(k_j - E(k_j)) f_{mn}(k_1, k_2 \dots k_{m-1}) = -np_ip_j \tag{1.77}$$

Esempio 1.11.1 *Sondaggio elettorale*
Un sondaggio elettorale eseguito su n =1000 persone assegna al partito A una percentuale p_A=42.7% dei voti, e una percentuale p_B=38.0% al partito B. Vogliamo conoscere l'incertezza statistica sul distacco $\delta = p_A - p_B$ previsto dal sondaggio.

Dalla formula di propagazione degli errori

$$\sigma^2(p_A - p_B) = \sigma^2(p_A) + \sigma^2(p_B) - 2cov(p_A, p_B) =$$
$$= \frac{p_A \cdot (1 - p_A)}{n} + \frac{p_B \cdot (1 - p_B)}{n} + 2\frac{p_A \cdot p_B}{n}$$

e dai dati possiamo dunque stimare

$$\delta \simeq (4.7 \pm 2.8)\%$$

1.12 Distribuzione di Poisson

Consideriamo ora il caso di un esperimento di conteggio in cui il numero di "successi" k non sia limitato, ovvero $k \ll n$. Se contiamo ad esempio il numero di stelle cadenti osservate durante una notte estiva, avremo tipicamente conteggi di poche unità, ma non vi è un limite fisico al numero di meteore che possono provocare un fenomeno visibile. Questo tipo di eventi può essere descritto da una distribuzione binomiale in cui $n \to \infty, p \to 0$ ma $\lambda = pn$ ha un valore finito. In questo limite

$$\begin{aligned} f_b(k; n, p) &= \frac{n!}{k!(n-k)!} \left(\frac{\lambda}{n}\right)^k \left(1 - \frac{\lambda}{n}\right)^{n-k} = \\ &= \frac{\lambda^k}{k!} \left(1 - \frac{\lambda}{n}\right)^n \frac{n(n-1)\dots(n-k+1)}{n^k} \frac{1}{\left(1 - \frac{\lambda}{n}\right)^k} \end{aligned} \tag{1.78}$$

tende alla **distribuzione di Poisson**

$$f_P(k;\lambda) = \frac{\lambda^k}{k!}e^{-\lambda} \tag{1.79}$$

che dipende quindi dal solo parametro λ.

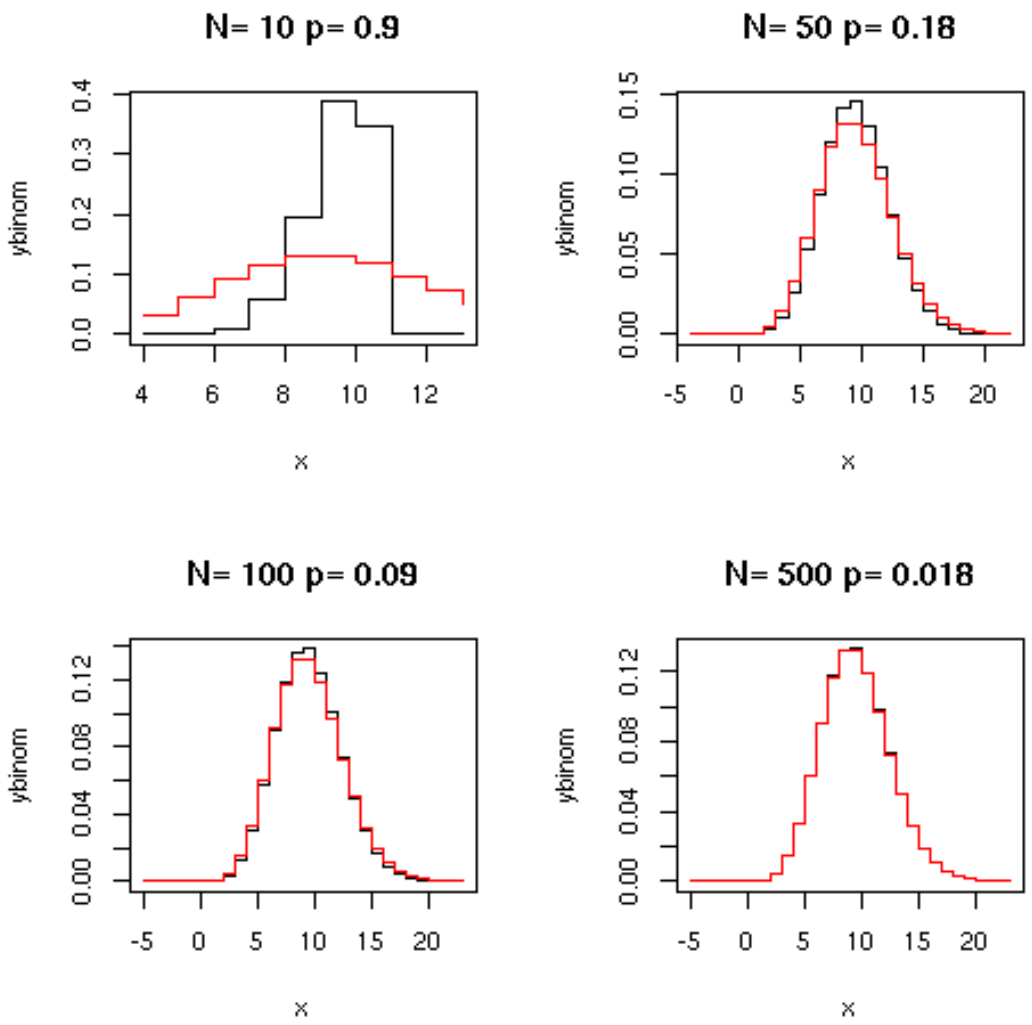
Il valore atteso $E(k) = \lambda$ e la varianza $\sigma^2(k) = \lambda$ si ottengono dal limite dei corrispondenti valori per la binomiale o usando la 1.66.
E' interessante notare che anche il valore del momento terzo è pari a λ e dunque l'asimmetria

$$y = \frac{\sum_{i=0}^{\infty}(k-\lambda)^3 f_P(k;\lambda)}{\lambda^{3/2}} = \lambda^{-1/2} \tag{1.80}$$

diminuisce al crescere di λ.

Il seguente codice permette di visualizzare il confronto fra la distribuzioni binomiale e poissoniana con $\lambda = N \cdot p$

```
binpois <-function(n=10000,p=0.001) {
 lambda=n*p
 sd=sqrt(n*p*(1-p))
# scegliamo i limiti del plot come valore atteso
# +/- 5 deviazioni standard
 x=as.integer(lambda-5*sd):as.integer(lambda+5*sd)

 ybinom=dbinom(x,size=n,prob=p)
 ypois=dpois(x,lambda=lambda)

 plot(x,ybinom,type="s", main=paste("N=",n,"p=",p))
 points(x,ypois,type="s",col="red")
}

par(mfrow=c(2,2))
binpois(10,0.9)
binpois(50,0.18)
binpois(100,0.09)
binpois(500,0.018)
```



1.13 Limite della distribuzione di Poisson, teorema del limite centrale

La somma di due variabili k_1 e k_2 che seguono distribuzioni binomiali $f_b(k_1; n_1, p)$ e $f_b(k_2; n_2, p)$ sarà

$$k_s = k_1 + k_2 = \sum_{i=1}^{n_1+n_2} B_i$$

e seguirà quindi una distribuzione binomiale $f_b(k_s; n_1 + n_2, p)$.
Analogamente, la somma di due variabili poissoniane con parametri λ_1 e λ_2 segue ancora una distribuzione poissoniana $f_P(k, \lambda_1 + \lambda_2)$. Se sommiamo N variabili poissoniane con parametro λ_0 otteniamo ancora una poissoniana con parametro $N\lambda_0$, la cui distribuzione al crescere di N è sempre più simmetrica e assume la caratteristica forma a campana. Si dimostra che il limite continuo della distribuzione di Poisson per $\lambda \to \infty$ non è altro che la distribuzione normale con

valore atteso e varianza pari a λ:

$$lim_{\lambda\to\infty}\ f_P(k;\lambda) = \frac{1}{\sqrt{2\pi\lambda}} \exp \frac{-(k-\lambda)^2}{2\lambda} \tag{1.81}$$

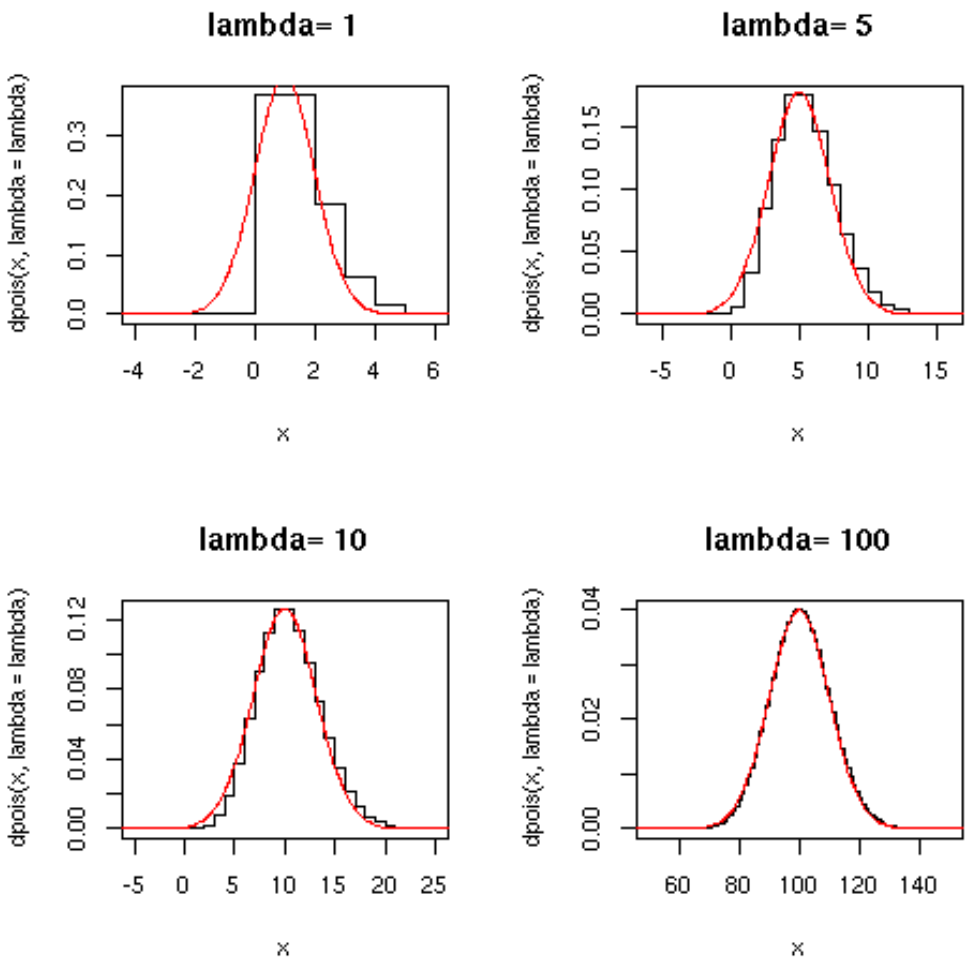


Figura 1.3: Confronto fra distribuzioni di Poisson $f_P(x;\lambda)$ e di Gauss $\phi_G(x;\mu=\lambda,\sigma^2=\lambda)$ per vari valori di λ

Il confronto visivo fra la distribuzione di Poisson e il suo limite asintotico è ottenuto dal seguente codice

```
gauspois <-function(lambda) {
 sd=sqrt(lambda)
 x=as.integer(lambda-5*sd):as.integer(lambda+5*sd)
 plot(x,dpois(x,lambda=lambda),type="s",
   main=paste("lambda=",lambda))
 curve(dnorm(x,mean=lambda,sd=sd),add=T,col="red")
}
```

```
par(mfrow=c(2,2))
gauspois(1)
gauspois(5)
gauspois(10)
gauspois(100)
```

Il fatto che la somma di infiniti termini poissoniani risulti in una distribuzione gaussiana è un caso particolare del **teorema del limite centrale**:
la funzione densità della somma di N variabili x_i identicamente distribuite con valore atteso a e varianza s^2, tende, nel limite $N \to \infty$, alla distribuzione normale con $\mu = Na, \sigma^2 = Ns^2$, indipendentemente dalla forma della distribuzione $d(x)$[3]

Esempio 1.13.1 *Convoluzione di più variabili distribuite uniformemente*
Per dimostrare l'importanza pratica del teorema del limite centrale, confrontiamo la distribuzione della somma di n variabili distribuite uniformemente nell'intervallo $[-1, 1]$. Come visto in precedenza, la distribuzione per $n = 2$ è una funzione triangolare. Più in generale, la funzione densità è descritta da polinomi di grado $n - 1$ e solo nel limite $n \to \infty$ diventa una gaussiana. Tuttavia, per un valore finito del numero di osservazioni N, la distribuzione può essere descritta da una gaussiana con ottima approssimazione già per valori di n molto bassi:

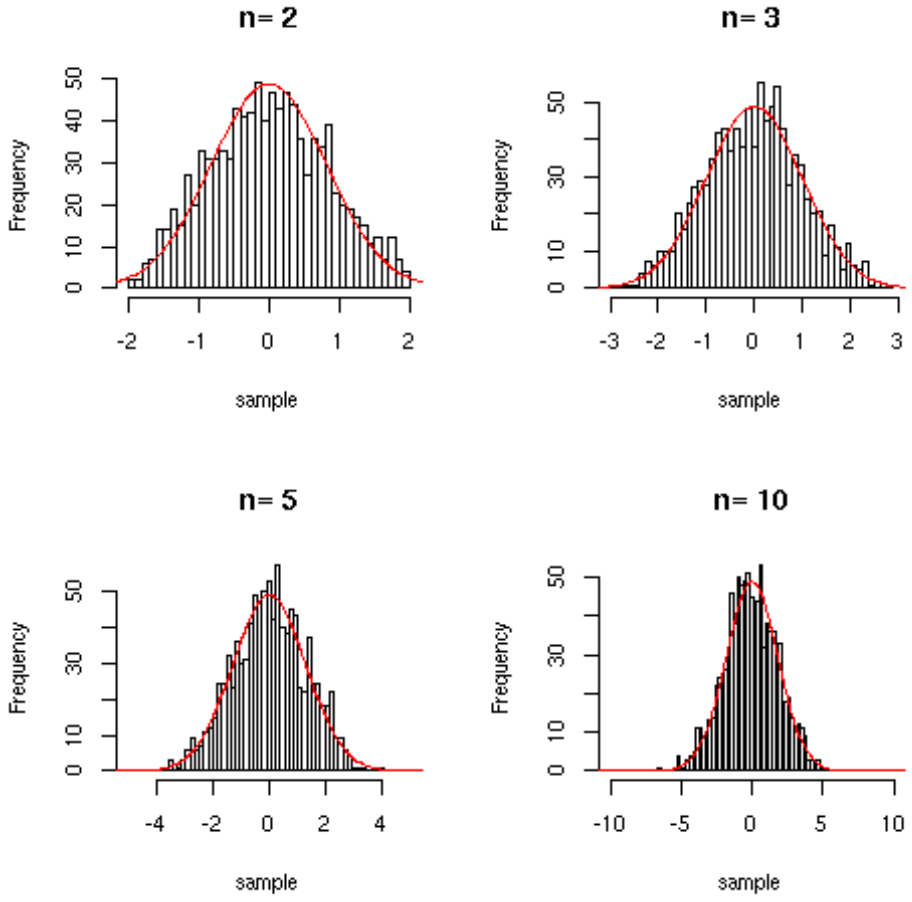
```
centrallimit <-function(n,N=5000) {
 sample = runif(N,-1,1)
 if (n >1) {
  for (i in 1:(n-1)) {
   sample = sample + runif(N,-1,1)
  }
 }
 delta=sqrt(2*n)/20
# confrontiamo la distribuzione ottenuta con la gaussiana
# che ci aspettiamo dal teorema del limite centrale
 hist(sample,breaks=seq(-1.*n,1.*n,delta),main=paste("n=",n))
 curve(dnorm(x,sd=2*sqrt(n/12))*N*delta,add=T,col="red")
}

par(mfrow=c(2,2))
centrallimit(n=2)
centrallimit(n=3)
centrallimit(n=5)
```

[3] si veda [1] per la dimostrazione. Il teorema può essere esteso al caso di variabili non identicamente distribuite, a patto che sia valida la seguenta condizione (detta di Lindeberg):

$$\lim_{n\to\infty} \sum_{i=1}^{n} \mathrm{E}\left(\frac{(x_i - E(x_i))^2}{\sum_{i=1}^{n} \sigma_i^2} \middle| |x_i - E(x_i)| > \varepsilon \sqrt{\sum_{i=1}^{n} \sigma_i^2} \right) = 0 \quad \forall \varepsilon > 0$$

dove per $E(X|Y > c)$ si intende il valore atteso della variabile che ha valore X per $Y > c$ e 0 altrimenti.

```
centrallimit(n=10)
```



Di norma, gli strumenti di misurazione sono soggetti ad un gran numero di piccoli effetti che possono perturbare la misura. La fisica fondamentale di questi effetti può portare alle distribuzioni di probabilità più diverse; tuttavia, grazie al principio di sovrapposizione, spesso l'effetto totale risulta dalla somma dei singoli effetti. In tal caso il teorema del limite centrale ci permette di poter assumere che gli errori seguano la distribuzione di Gauss, che non a caso è chiamata "normale".

Bisogna comunque tener presente che la distribuzione normale non discende da una legge fisica fondamentale, ma da un teorema matematico valido solo nel limite di infiniti contributi all'errore. Nei casi reali possiamo dunque aspettarci che l'assunzione di errori gaussiani sia valida entro una certa approssimazione, ma praticamente mai rigorosamente. Quello che conta è se la deviazione dalla normalità sia significativa rispetto all'incertezza statistica del nostro campione di dati (cfr. eq. 1.75). Nell'esempio precedente, nel caso di $n = 10$ variabili e un campione di dimensione $N = 5000$, la distribuzione empirica dei dati non mostra deviazioni significative dall'ipotesi normale[4], ma aumentando la dimensione del campione (ad esempio

[4]il numero di dati in ciascun intervallo dell'istogramma k_i è una variabile aleatoria che segue la distribuzione multinomiale con deviazione standard $\sigma_i = \sqrt{Np_i(1-p_1)} \simeq \sqrt{k_i}$. Nel grafico del caso $n = 10$ le deviazioni fra i

$N = 500000$), l'approssimazione normale non sarebbe più giustificata per descrivere il campione di dati.
Non è infrequente che una distribuzione, a prima vista gaussiana, nasconda code non gaussiane; questo può avvenire, ad esempio, quando la probabilità di discostarsi dal valore atteso di oltre 3 deviazioni standard sia dominata da un singolo effetto fisico (tipicamente non lineare).

1.14 La distribuzione di Pearson

Se $x_i, i = 1, \ldots n$ sono variabili aleatorie gaussiane indipendenti con valore atteso μ_i e deviazione standard σ_i, la variabile

$$\chi^2 = \sum_{i=1}^{n} \frac{(x_i - \mu_i)^2}{\sigma_i^2} \tag{1.82}$$

che rappresenta la somma dei quadrati degli scarti rispetto al valore atteso in unità di deviazione standard, segue una distribuzione, detta di Pearson o di χ^2 ("chi quadro"), che dipende dal solo parametro $g = n$:

$$f_{\chi^2}(\chi^2; n) = \frac{1}{2^{g/2}\Gamma(g/2)} (\chi^2)^{g/2-1} \exp(-\chi^2/2) \tag{1.83}$$

dove

$$\Gamma(x) = \int_0^\infty e^{-t} t^{x-1} dt \tag{1.84}$$

è la funzione gamma di Eulero. La distribuzione χ^2 è un caso particolare della distribuzione Gamma. Il parametro g è detto "numero di gradi di libertà" per motivi che saranno più chiari nei prossimi capitoli. Questa proprietà ha importanti applicazioni nell'analisi statistica, come vedremo nei capitoli 3 e 4.

Dalla definizione di varianza, notiamo che il valore atteso di ciascun membro della sommatoria è 1, e dunque il valore atteso della distribuzione è pari a g. Si dimostra che la varianza è pari a $2g$, e il momento terzo è $8g$. La larghezza relativa della distribuzione $\sigma(x)/E(x) = \sqrt{2/g}$ e l'asimmetria $\mu_3/\sigma^{3/2} = 2\sqrt{2/g}$ diminuiscono dunque entrambe al crescere di g.
Nel software R la PDF della distribuzione χ^2 può essere ottenuta dalla funzione dchisq()

valori k_i e i valori predetti nell'ipotesi normale sono compatibili con questa deviazione standard, come vedremo meglio nel paragrafo 3.2.

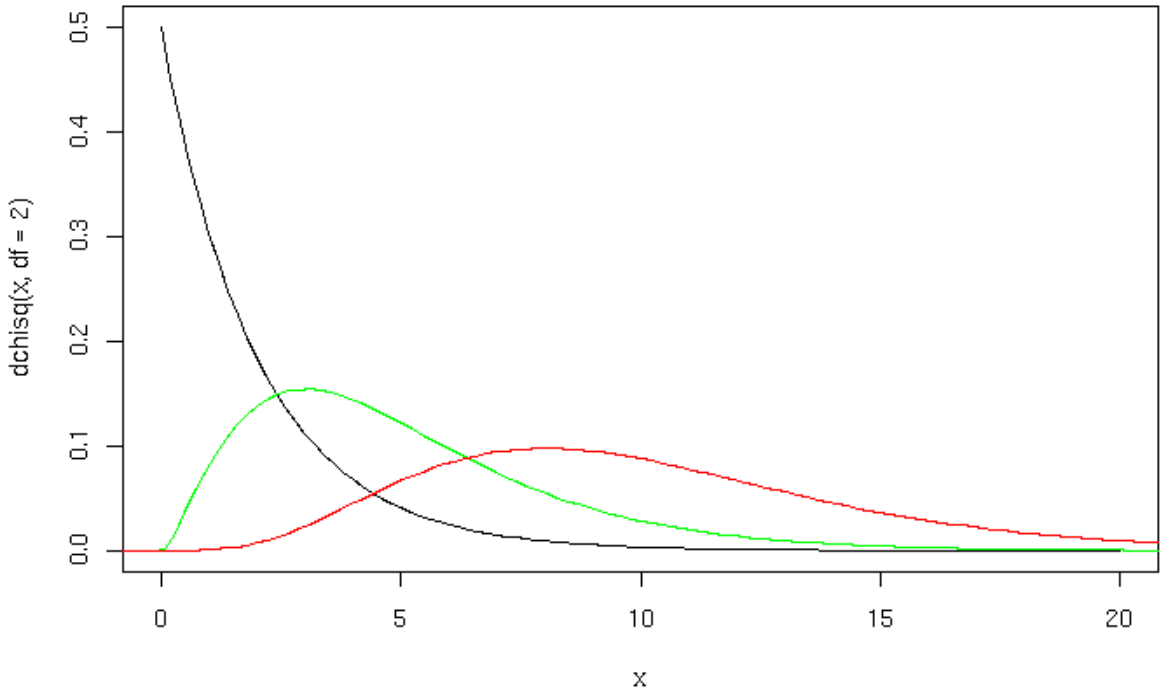


Figura 1.4: Distribuzione di Pearson per 2, 5, 10 gradi di libertà. [output del comando *curve(dchisq(x, df=2),0, 20) ; curve(dchisq(x, df=5),add=T,col="green") ; curve(dchisq(x, df=10),add=T,col="red")*]

Capitolo 2

Campionamento e Stimatori

Invertiremo ora la prospettiva del capitolo precedente, mettendoci dal punto di vista dello sperimentatore che dispone di un campione finito di dati da cui si vogliono ricavare informazioni sul processo stocastico che li ha generati. Il nostro generico campione

$$\mathcal{S} = (\underline{x}_1, \underline{x}_2, \ldots, \underline{x}_N)$$

è costituito da N osservazioni di un generico set di variabili, continue e/o discrete, rappresentate dal vettore $\underline{x}$. Nell'ipotesi di misura riproducibile, assumeremo che la funzione densità di probabilità di $\underline{x}$ sia la stessa per tutte le misure

$$d_i(\underline{x}_i) = d(\underline{x}_i) \qquad \forall i \tag{2.1}$$

e che le osservazioni siano indipendenti, ovvero

$$d(\mathcal{S}) = \prod_{i=1}^{N} d(\underline{x}_i) \tag{2.2}$$

Lo scopo principale dell'analisi statistica è quello di stimare $d(\underline{x})$, o, più comunemente, i suoi parametri, a partire dal campione $\mathcal{S}$. La dimensione finita del campione limita in generale la precisione della stima, a cui sarà dunque associata una incertezza statistica.

2.1 Stimatori

Uno **stimatore** è una funzione $\overline{\theta}(\mathcal{S})$ del campione di dati che permette di stimare un parametro θ della legge di probabilità $d(\underline{x})$, ad esempio il suo valore atteso o la sua varianza. A differenza di θ, che è una grandezza con un valore univoco, lo stimatore è una variabile aleatoria, a cui possiamo associare una funzione densità di probabilità nello spazio dei campioni.

Consideriamo il caso della misura di una singola variabile reale x. La **media** aritmetica dei valori

$$\overline{x} = \frac{\sum_{i=1}^{N} x_i}{N} \tag{2.3}$$

è un esempio di stimatore del valore atteso di x. Infatti

$$E(\overline{x}) = \frac{\sum_{i=1}^{N} E(x_i)}{N} = E(x) \tag{2.4}$$

Si dice in tal caso che lo stimatore è corretto (unbiased), poiché il suo **bias** (distorsione), definito in generale come

$$b(\overline{\theta}) = E(\overline{\theta}) - \theta \tag{2.5}$$

è nullo.
La deviazione standard dello stimatore indica di quanto la stima può differire dal suo valore atteso, che è, per uno stimatore corretto, il valore a cui siamo interessati. Può dunque essere interpretata come errore della stima. Nel caso della media

$$\sigma(\overline{x}) = \sqrt{\frac{\sum_{i=1}^{N} \sigma^2(x_i)}{N^2}} = \sqrt{\frac{N\sigma^2(x)}{N^2}} = \frac{\sigma(x)}{\sqrt{N}} \tag{2.6}$$

dove abbiamo usato l'assunzione 2.2 di indipendenza delle misure. La media aritmetica è un esempio di stimatore **consistente**, per cui vale

$$lim_{N\to\infty}\ \sigma(\overline{\theta}) = 0 \tag{2.7}$$

Un altro parametro importante da stimare è la varianza di x, che ci serve, se non altro, per stimare l'errore sulla media. Essendo la varianza il valore atteso dello scarto quadratico rispetto al valore atteso, possiamo pensare di usare la media aritmetica dello scarto quadratico rispetto al valore stimato di $E(x)$:

$$\overline{\sigma_0^2} = \frac{\sum_{i=1}^{N}(x_i - \overline{x})^2}{N} \tag{2.8}$$

Tuttavia questo stimatore non risulta essere corretto:

$$\begin{aligned}
E(\overline{\sigma_0^2}) &= \frac{1}{N} E\left[\sum_i (x_i - \overline{x})^2\right] = \\
&= \frac{1}{N} E\left[\sum_i \left((x_i - E(x)) - (\overline{x} - E(x))\right)^2\right] = \\
&= \sigma^2(x) + \sigma^2(\overline{x}) - \frac{2}{N} E\left[(\overline{x} - E(x)) \sum_i (x_i - E(x))\right] = \\
&= \sigma^2(x) - \sigma^2(\overline{x}) = \sigma^2(x)\frac{(N-1)}{N}
\end{aligned} \tag{2.9}$$

e lo stimatore corretto della deviazione standard è dunque

$$\overline{\sigma} = \sqrt{\frac{N}{(N-1)}\overline{\sigma_0^2}} = \sqrt{\frac{\sum_{i=1}^{N}(x_i - \overline{x})^2}{N-1}} \tag{2.10}$$

che è comunemente chiamato **scarto quadratico medio** (o rms, dall'inglese *root mean square*)[1]

La varianza di $\overline{\sigma}$ può essere calcolata come

$$\sigma^2(\overline{\sigma^2}) = \frac{1}{N}\left(\mu_4 - \frac{N-3}{N-1}\sigma^4\right) \tag{2.11}$$

da cui, usando la formula di propagazione degli errori e sostituendo i momenti con i loro stimatori, si ottiene

$$\overline{\sigma(\overline{\sigma})} \simeq \frac{1}{2\overline{\sigma}}\sqrt{\frac{\overline{\mu_4} - \overline{\sigma}^4}{N}} \tag{2.12}$$

dove

$$\overline{\mu_4} = \frac{\sum_{i=1}^{N}(x_i - \overline{x})^4}{N-1}$$

Se x è una variabile normale, o approssimativamente tale, si ha $\mu_4 = 3\sigma^4$ da cui

$$\overline{\sigma(\overline{\sigma})} \simeq \frac{\sigma(\overline{\sigma^2})}{2\sigma} \simeq \frac{1}{2\sigma}\sqrt{\frac{2\sigma^4}{N}} \simeq \frac{\sigma}{\sqrt{2N}} \tag{2.13}$$

dove si è usata la formula di propagazione degli errori per passare da $\sigma(\overline{\sigma^2})$ a $\sigma(\overline{\sigma})$.
Analogamente al caso della varianza, lo stimatore corretto per la covarianza risulta essere

$$\overline{cov(X,Y)} \simeq \frac{\sum_i (x_i - \overline{x})(y_i - \overline{y})}{N-1} = \frac{N}{N-1}(\overline{xy} - \overline{x}\cdot\overline{y}) \tag{2.14}$$

Per la stima del coefficiente di correlazione si usa normalmente

$$\overline{\rho} = \frac{\overline{cov(X,Y)}}{\overline{\sigma_X}\ \overline{\sigma_Y}} \tag{2.15}$$

nonostante non risulti essere uno stimatore corretto. Nel caso in cui le variabili X e Y seguono una distribuzione normale bivariata si ha

$$E(\overline{\rho}) = \rho - \frac{\rho(1-\rho^2)}{2N} + o(N^{-2}) \tag{2.16}$$

Per semplicità la correzione al bias non viene normalmente applicata, essendo comunque uno stimatore "asintoticamente corretto" (il bias tende a zero nel limite $N \to \infty$). L'errore associato, sempre nel caso normale, è

$$\sigma(\overline{\rho}) \simeq \frac{(1-\overline{\rho}^2)}{\sqrt{N}} \tag{2.17}$$

La procedura per la stima dei quantili di una variabile continua x consiste nell'ordinare i valori in senso crescente ed utilizzare il valore corrispondente alla frazione di valori voluta. Ad esempio, per stimare la mediana, prenderemo il valore centrale della lista se il numero N di valori è dispari, o la media dei due valori centrali se N è pari. Il bias e la varianza di questa stima dipendono dalla distribuzione di x.

[1] Possiamo spiegare intuitivamente il fattore $N-1$ col fatto che lo scarto quadratico empiricamente non può essere calcolato rispetto al valore vero del valore atteso, ma solo rispetto al suo valore stimato, che, essendo ricavato dagli stessi valori x_i, tende a sottostimare le variazioni. Si dice in questo caso che per il calcolo della media abbiamo usato un grado di libertà del sistema, e dunque ne restano $N-1$ per la stima della varianza.

2.2 Statistica descrittiva

Viene chiamata "statistica descrittiva" l'insieme di tecniche che permettono di sintetizzare, tramite semplici stimatori o tramite visualizzazione grafica, le caratteristiche di un campione di dati. Gli stimatori visti nel paragrafo precedente forniscono una prima indicazione riassuntiva della distribuzione dei valori e delle dipendenze fra variabili reali. Tutti i software di analisi dati forniscono strumenti di statistica descrittiva più o meno evoluti.

Esempio 2.2.1 *Esplorazione di un campione di dati*
I risultati di misure sperimentali vengono solitamente registrati sotto forma di tabelle. Nel software R, la classe più idonea a rappresentare un generico campione è il *dataframe*, che contiene una riga per ciascuna osservazione e una colonna per ciascuna variabile, la quale può essere rappresentata da un qualunque tipo di classe (tipicamente *numeric* per variabili continue e *integer* o *factor* per variabili discrete). Consideriamo ad esempio uno dei dataframes presenti di default nel software, chiamato *Teoph*, che contiene il risultato di un test clinico su un farmaco anti–asma. Digitando il nome del dataframe viene visualizzata l'intera tabella, mentre i singoli valori possono essere richiamati con la notazione *Teoph[riga, colonna]*. Il comando *summary(Teoph)* riassume i valori del dataframe. Per variabili reali, sono mostrati i valori degli stimatori di media, mediana, quartili. Possiamo utilizzare le funzioni *mean()* e *sd()* per calcolare la media e la rms di una variabile, ad esempio i comandi

```
> mean(Theoph[Theoph$Time==0,'Wt'])
[1] 69.58333
> sd(Theoph[Theoph$Time==0,'Wt']) / sqrt( nrow(Theoph[Theoph$Time==0,]) )
[1] 2.743307
```

permettono di calcolare la media del peso dei soggetti all'inizio della somministrazione ed il suo errore (cfr eq. 2.6).
Le funzioni $var()$, $cov()$ e $cor()$ forniscono le stime di varianza, covarianza e coefficiente di correlazione, e possono essere applicate anche ad un dataframe per ottenere una stima della matrice varianza/covarianza o di correlazione. Poiché queste funzioni sono definite solo per variabili continue, nel nostro esempio possiamo usarle escludendo la prima colonna del dataframe che è una variabile discreta:

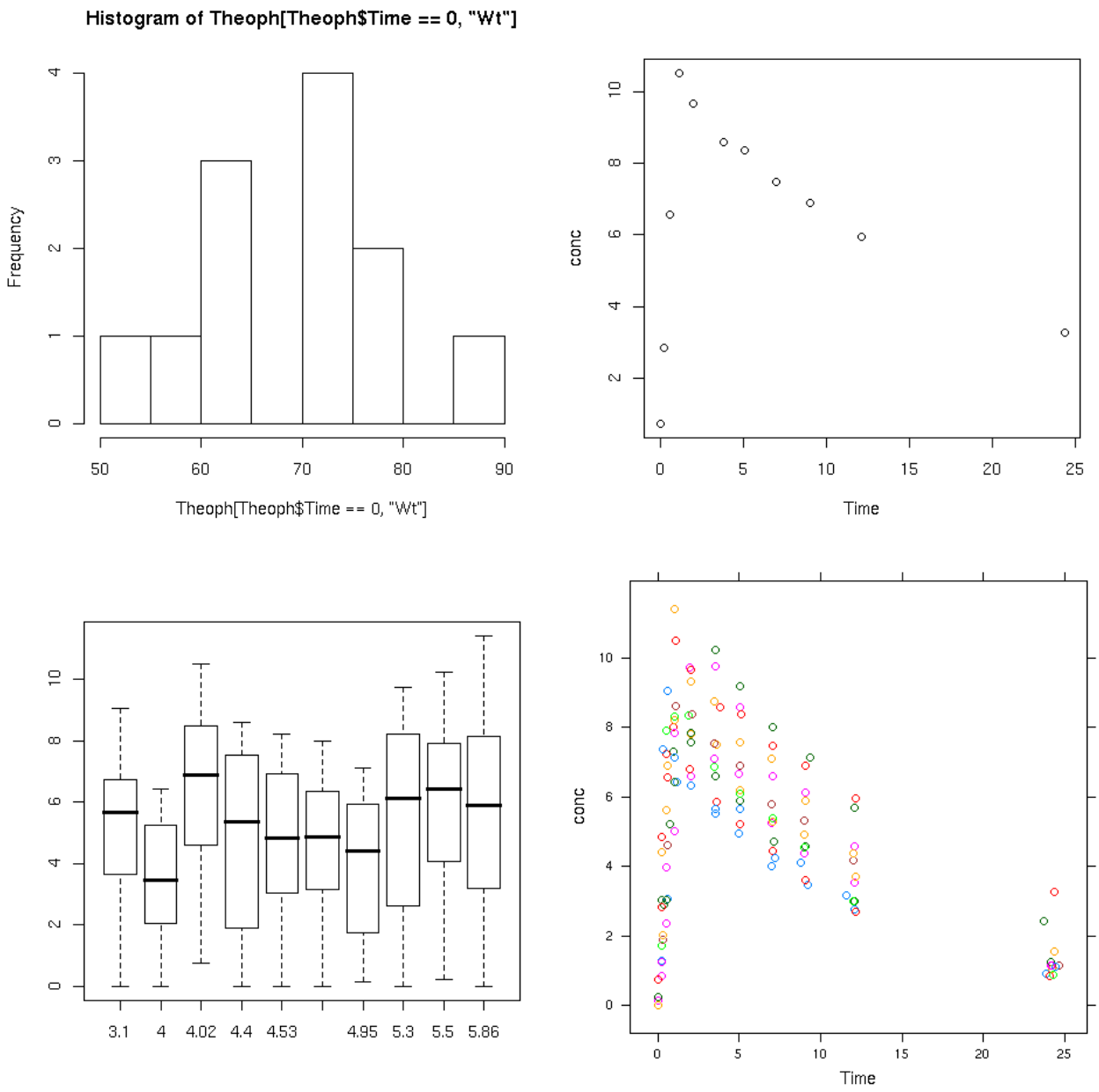
```
> cov(Theoph[,-1])
              Wt          Dose          Time       conc
Wt   83.41498728 -6.4902379135  0.0262989822 -1.8797099
Dose -6.49023791  0.5156305980 -0.0004958651  0.1584859
Time  0.02629898 -0.0004958651 47.9688158860 -6.0291823
conc -1.87970992  0.1584858779 -6.0291822693  8.2215204
> cor(Theoph[,-1])
                Wt          Dose          Time        conc
Wt    1.0000000000 -0.9896217523  0.0004157549 -0.07177823
Dose -0.9896217523  1.0000000000 -0.0000997045  0.07697420
Time  0.0004157549 -0.0000997045  1.0000000000 -0.30360075
conc -0.0717782261  0.0769741957 -0.3036007533  1.00000000
```

L'**istogramma** è il modo più comune di rappresentare graficamente la distribuzione dei valori di una variabile:

```
hist(Theoph[Theoph$Time==0,'Wt'])
```

mentre funzioni quali $plot(), boxplot(), xyplot()$[2] possono essere usate per mettere in evidenza mutue dipendenze fra le variabili, ad esempio:

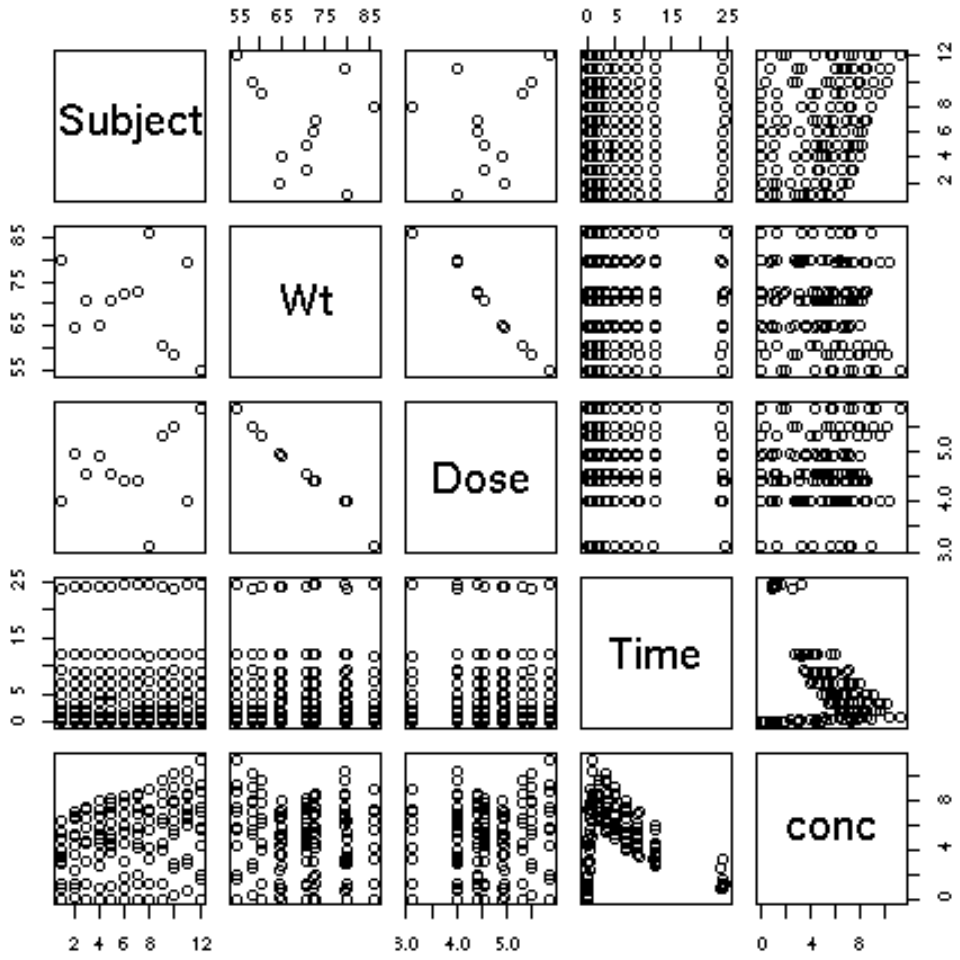
```
plot( conc ~ Time, data=Theoph[Theoph$Subject==1,])
boxplot(conc ~ Dose, data=Theoph)
library(lattice)
xyplot( conc ~ Time, data=Theoph, groups=Subject)
```



[2]si usi il comando $help(nome\_funzione)$ per avere dettagli su ciascuna funzione. La funzione $xyplot()$ richiede il pacchetto *lattice*.

Il comando *pairs*() permette di visualizzare le mutue dipendenze fra ogni coppia di variabili, una sorta di versione grafica della matrice di correlazione:

```
pairs(Theoph)
```



2.2.1 Cifre significative

Ê importante sottolineare come qualunque stima sia soggetta ad un errore statistico dovuto alla dimensione finita del campione. Nel riportare il valore di uno stimatore $\overline{\theta}$ bisogna dunque tenere presente che l'informazione sul parametro θ è contenuta solo nelle cifre, dette significative, che sono dello stesso ordine o maggiori dell'errore $\sigma(\overline{\theta})$. Ad esempio, nel calcolo della media del peso nell'esempio 2.8.1

```
> mean(Theoph[Theoph$Time==0,'Wt'])
[1] 69.58333
```

```
> sd(Theoph[Theoph$Time==0,'Wt']) / sqrt ( nrow(Theoph[Theoph$Time==0,]) )
[1] 2.743307
```

vi sono solo 2 cifre significative, e il risultato della stima deve essere riportato come 70 ± 3, o al massimo come 69.6 ± 2.7 se si vogliono considerare due cifre significative per l'errore (cosa che può essere opportuna in misure di precisione se la prima cifra significativa dell'errore è 1 o 2). Le cifre che seguono non portano evidentemente alcuna informazione.
Riportare le sole cifre significative è ancora più importante nel caso (frequente al di fuori delle pubblicazioni scientifiche) in cui il risultato di una misura sia dato senza citare esplicitamente l'errore. Ad esempio, se riportassimo la nostra stima come 69.58333 (copiando banalmente il numero stampato da R), daremmo al lettore l'impressione di aver effettuato una misura accurata fino alla quinta cifra decimale, cosa evidentemente falsa.

Esempio 2.2.2 *Stima del rate di un evento*
Vogliamo stimare la probabilità per unità di tempo (*rate*) che una meteorite con massa maggiore di 1 Kg colpisca la superficie terrestre. Disponiamo di una serie di misure del tempo trascorso fra due eventi consecutivi (in giorni), elencati nel file
/afs/math.unifi.it/service/Rdsets/decaytimes

Il tempo fra due eventi, assumendo che questi siano indipendenti, seguirà la distribuzione esponenziale

$$d(t) = re^{-rt}$$

il cui valore atteso è $\tau = 1/r$. Stimiamo dunque τ tramite la media aritmetica e calcoliamo r ed il suo errore tramite la formula di propagazione degli errori:

```
dati = scan("/afs/math.unifi.it/service/Rdsets/decaytimes")
tau = mean(dati)
sigma.tau = sd(dati)/sqrt( length(dati) )
r = 1/tau
sigma.r = sigma.tau / tau^2
cat("r=",r," +/- ",sigma.r,"\n")
```

Il risultato è $r = (0.31 \pm 0.01)$ giorni^{-1}.

Esempio 2.2.3 *Conteggi di raggi cosmici*
Un esperimento situato presso l'università di Stanford misura continuamente il flusso di particelle cosmiche (che a livello del mare sono costituiti essenzialmente da particelle chiamate muoni). Il flusso nella direzione verticale viene misurato osservando le particelle che attraversano due rivelatori di superficie S posti uno sopra l'altro a una distanza d.
I dati reali sono pubblicati in rete all'indirizzo
http://www2.slac.stanford.edu/vvc/cosmicrays/crdatacenter.html

Dai conteggi ottenuti in intervalli di 2 minuti tabulati nel file
/afs/math.unifi.it/service/Rdsets/CosmoData.txt

si vuole stimare il flusso di raggi cosmici nella direzione verticale, definito come numero di particelle per unità di tempo, superficie e angolo solido

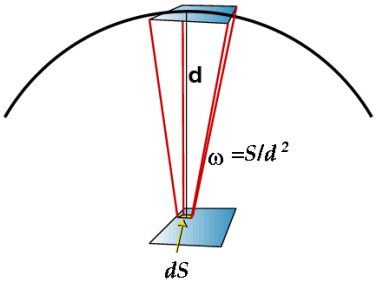
$$F_\mu(\theta = \pi/2) = \frac{d^3 N_\mu}{dt d\omega dS}$$

sapendo che $d = 47$ cm e $S = 251.7$ cm2, e che la particelle che attraversano i rivelatori possono essere considerate eventi indipendenti.

Data l'indipendenza degli eventi, il numero di conteggi C misurato in ogni intervallo seguirà una distribuzione di Poisson con valore atteso pari a

$$E(C) = \Delta T \cdot \frac{dC}{dt} = \Delta T \int_{S,\omega} F_\mu d\omega dS \simeq \Delta T F_\mu(\theta = \pi/2) \frac{S}{d^2} S$$

(abbiamo trascurato le dimensioni lineari dei rivelatori rispetto a d, approssimando l'angolo azimutale θ a $\pi/2$ per tutti gli eventi e l'angolo solido accettato per ciascun punto del rivelatore a $\omega = S/d^2$)



F_μ è dunque semplicemente proporzionale al valore atteso della distribuzione, che può essere stimato dalla media aritmetica

$$\overline{C} = \sum_i C_i / N$$

la cui deviazione standard è stimata da

$$\overline{\sigma_C} = \sqrt{\frac{1}{N} \frac{\sum_i (C_i - \overline{C})^2}{(N-1)}}$$

Il seguente codice esegue il calcolo:

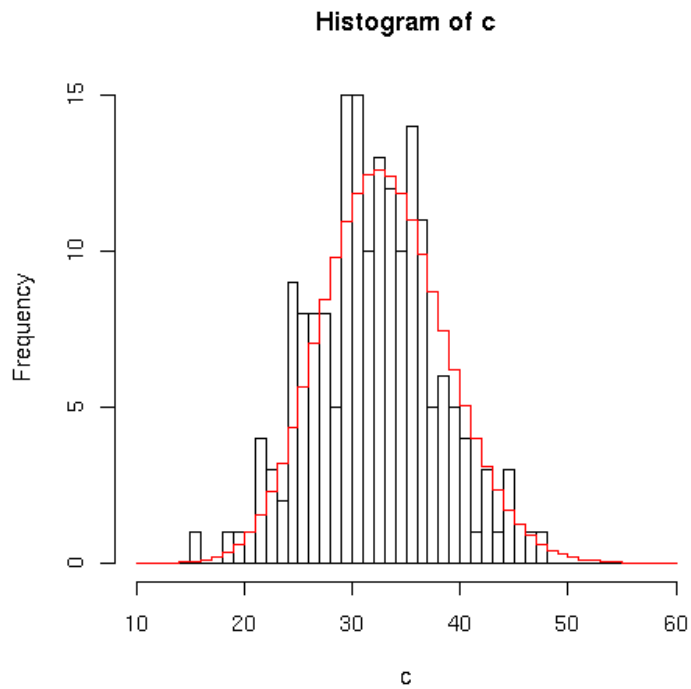
```
# creiamo un dataframe dal file
# l'opzione skip permette di saltare le prime righe
# di spiegazione nel file
df =read.table(file="/afs/math.unifi.it/service/Rdsets/CosmoData.txt",
           skip=7)
# i valori dei conteggi sono la terza colonna del dataframe
c = df[,3]
n=length(c)
d=47     # cm
S=251.7  # cm^2
deltaT=2 # minuti

cm=mean(c)
dcm=sqrt(var(c)/n)
cat("average counting is ",cm," +/- ",dcm,"\n")
k=(d/S)^2/deltaT
cat("il flusso è ",cm*k," +/- ",dcm*k," muons/cm^2/sterad/min\n")
```

otteniamo $F_\mu = (0.566 \pm 0.007)$ muoni/(cm^2 sterad min)

Naturalmente avremmo potuto considerare la somma di tutti i conteggi come un unico conteggio corrispondente ad un intervallo $\Delta T' = N\Delta T$. In tal caso la media coincide con l'unico valore misurato $C' = \sum_i C_i$ e il risultato per il flusso resta invariato. Sapendo che la varianza della distribuzione di Poisson è uguale al valore atteso, il valore C' è anche uno stimatore della varianza. L'errore standard su C' è dunque semplicemente $\sqrt{C'}$.

L'aver suddiviso i conteggi in N intervalli ci permette comunque di valutare visivamente la correttezza dell'ipotesi poissoniana:

```
x=10:60
hist(c,breaks=x)
points(x,
  dpois(x,lambda=cm)*n,
  type="s",col="red")
```



2.3 Il principio di massima verosimiglianza

Il principio di **massima verosimiglianza** (ML, dall'inglese maximum likelihood) fornisce un criterio generale per determinare uno stimatore di un parametro θ. La funzione di *likelihood*, nel caso in cui vogliamo stimare un singolo parametro θ, è definita come

$$\mathcal{L}(\mathcal{S};\theta') = \prod_{i=1}^{N} f(\underline{x}_i|\theta = \theta') \tag{2.18}$$

e dunque non è altro che la densità di probabilità del campione, pensata come funzione di θ. La funzione risponde quindi alla domanda: quanto sarebbe probabile osservare il nostro campione se il parametro θ avesse il valore θ'? Chiaramente, valori di $\mathcal{L}$ più alti corrispondono a valori di θ più compatibili col campione osservato, e dunque più verosimili. Il principio di ML consiste nel prendere come stimatore di θ il valore $\overline{\theta}_{ML}$ che massimizza la funzione $\mathcal{L}$:

$$\mathcal{L}(\mathcal{S};\overline{\theta}_{ML}) \geq \mathcal{L}(\mathcal{S};\theta') \quad \forall\theta' \tag{2.19}$$

Il principio può essere esteso al caso di m parametri rappresentati dal vettore $\underline{\theta}$, nel qual caso la funzione $\mathcal{L}(\mathcal{S};\underline{\theta})$ va massimizzata nello spazio m–dimensionale dei parametri.

Come esempio, consideriamo il caso di un campione gaussiano:

$$\mathcal{L}(\mathcal{S};\mu,\sigma) = \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-(x_i-\mu)^2}{2\sigma^2}} \tag{2.20}$$

Per ricavare lo stimatore ML di μ, cerchiamo il massimo di $\mathcal{L}$ in funzione di μ tramite gli zeri della derivata parziale. In pratica è consigliabile cercare il massimo della funzione $\log \mathcal{L}$, in modo da poter trasformare la produttoria in una sommatoria di logaritmi (approffitando anche del fatto che molte distribuzioni di probabilità di interesse pratico contengono termini esponenziali):

$$\begin{aligned} & \frac{\partial \log \mathcal{L}(\mathcal{S};\overline{\mu}_{ML})}{\partial \mu} = 0 \\ \Longrightarrow \quad & \sum_i \frac{2(x_i - \overline{\mu}_{ML})}{2\sigma^2} = 0 \\ \Longrightarrow \quad & \overline{\mu}_{ML} = \frac{\sum_i x_i}{N} \end{aligned} \tag{2.21}$$

Il principio giustifica dunque l'utilizzo della media aritmetica come stimatore del valore atteso μ. Notiamo che lo stimatore è indipendente dall'altro parametro σ. Ricaviamo dunque lo stimatore per σ, fissando μ al suo stimatore ML:

$$\begin{aligned} & \frac{\partial \log \mathcal{L}(\mathcal{S};\overline{\sigma}_{ML})}{\partial \sigma} = 0 \\ \Longrightarrow \quad & -\frac{1}{\overline{\sigma}_{ML}} + \sum_i \frac{(x_i - \overline{\mu}_{ML})^2}{\overline{\sigma}^3_{ML}} = 0 \\ \Longrightarrow \quad & \overline{\sigma}_{ML} = \sqrt{\frac{\sum_i (x_i - \overline{\mu}_{ML})^2}{N}} \end{aligned} \tag{2.22}$$

che coincide con lo stimatore 2.8, che sappiamo essere corretto solo asintoticamente. Infatti, il principio non garantisce che gli stimatori trovati non siano distorti.

Possiamo verificare facilmente che la media è uno stimatore di ML del valore atteso anche per le altre distribuzioni introdotte nel primo capitolo (uniforme, esponenziale, binomiale, di Poisson).

Un'altra importante applicazione del principio riguarda il problema di combinare diverse misure sperimentali della stessa quantità: supponiamo di disporre di n misure della variabile

x, ciascuna con un diverso errore normale σ_i. La stima ML del valore atteso $\mu = E(x)$ sarà

$$\begin{aligned} & \frac{\partial \log \left(\prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma_1} e^{\frac{-(x_i-\mu)^2}{2\sigma_1^2}} \right)}{\partial \mu} = 0 \\ \Longrightarrow \quad & \sum_i \frac{2(x_i - \overline{\mu}_{ML})}{2\sigma_i^2} = 0 \\ \Longrightarrow \quad & \overline{\mu}_{ML} = \frac{\sum_i w_i x_i}{\sum_i w_i} \end{aligned} \tag{2.23}$$

dove $w_i = 1/\sigma_i^2$. Il principio di ML giustifica dunque la nota **formula della media pesata**. La varianza di questo stimatore è

$$\sigma^2(\overline{\mu}_{ML}) = \frac{1}{(\sum_i w_i)^2} \sum_i w_i^2 \sigma^2(x_i) = \frac{1}{\sum w_i} \tag{2.24}$$

2.4 Altre proprietà degli stimatori

In generale si possono avere più stimatori per un dato parametro. La scelta sul più opportuno da usare dipende dalle proprietà desiderate. Abbiamo già introdotto la consistenza e la correttezza come proprietà ottimali di uno stimatore. Fra due stimatori corretti, conviene solitamente scegliere il più **efficiente**, ovvero quello che ha varianza minima.

Esempio 2.4.1 *Stime di media e mediana nel caso gaussiano*
Come esempio, possiamo confrontare la media aritmetica e lo stimatore della mediana (introdotto nel paragrafo 2.1) per un campione gaussiano. Poiché la distribuzione è simmetrica, la mediana coincide col valore atteso, e dunque si tratta di due stimatori, entrambi corretti, del valore atteso. La media aritmetica risulta tuttavia uno stimatore più efficiente, come possiamo verificare analiticamente o numericamente, tramite una semplice simulazione di N campioni di $simN$ dati distribuiti secondo la distribuzione gaussiana standard. Per ciascun campione calcoliamo la media e la mediana e stimiamo la loro media e la loro deviazione standard:

```
mediamediana = function(N=100, simN=10000) {
  medie=c()
  mediane=c()
  for (i in 1:simN) {
    sample=rnorm(N)
    medie[i]=mean(sample)
    mediane[i]=median(sample)
  }
  xpar=par(mfrow=c(2,1))
  hist(medie,breaks=seq(-.5,.5,length=50))
  cat("media di medie  =",mean(medie)," +/- ",sd(medie)/sqrt(simN),"\n")
  cat("sigma di medie  =",sd(medie)," +/- ",sd(medie)/sqrt(2*simN),"\n")
  hist(mediane,breaks=seq(-.5,.5,length=50))
  cat("media di mediane=",mean(mediane)," +/- ",
```

```
                sd(mediane)/sqrt(simN),"\n")
  cat("sigma di mediane  =",sd(mediane)," +/- ",
                sd(mediane)/sqrt(2*simN),"\n")
  par(xpar)

}
> mediamediana()
media di medie  =  0.0002  +/-  0.0014
sigma di medie  =  0.0997  +/-  0.0010
media di mediane= -0.0002  +/-  0.0017
sigma di mediane=  0.1237  +/-  0.0012
```

Come previsto, i valori sono compatibili con bias nulli nei due casi, mentre per la deviazione standard della mediana troviamo 0.124 ± 0.001, contro il valore 0.1 per la media.
L'utilizzo della mediana può essere comunque preferibile nel caso in cui il nostro campione contenga degli *outlier*, ovvero dei dati anomali che non seguono la distribuzione attesa. La mediana è infatti più **robusta**, ovvero meno sensibile agli *outlier*: il suo valore non cambia se, ad esempio, il valore minimo del campione dista 5 o 50 σ dal valore atteso, cosa evidentemente non vera per la media.

L'efficienza di uno stimatore è strettamente legata alla **sufficienza**, definita da

$$P(\mathcal{S}|\overline{\theta}) \text{ indipendente da } \theta \tag{2.25}$$

Uno stimatore sufficiente contiene dunque tutta l'informazione su θ contenuta nel campione $\mathcal{S}$. Vale in questo caso il **teorema di fattorizzazione di Fisher**[5]

$$\overline{\theta} \text{ sufficiente} \iff \mathcal{L}(\mathcal{S};\theta) = g(\mathcal{S};\overline{\theta})h(\overline{\theta},\theta) \tag{2.26}$$

La forma della funzione di likelihood ci permette quindi di riconoscere "a vista" se uno stimatore è sufficiente. Una conseguenza del teorema è che, se esiste uno stimatore sufficiente, il metodo di ML lo trova. Infatti, massimizzare $\mathcal{L}$ in funzione di θ, implica massimizare la funzione $h(\overline{\theta},\theta)$, e il risultato non potrà che essere lo stimatore sufficiente $\overline{\theta}$ o una sua funzione, che sarà comunque sufficiente.
Ad esempio nel caso della gaussiana visto in precedenza

$$\begin{aligned}\mathcal{L} = \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-(x_i-\mu)^2}{2\sigma^2}} &= \left(\frac{1}{\sqrt{2\pi}\sigma}\right)^N e^{\frac{-N\overline{\sigma}^2_{ML}}{2\sigma^2}} \\ &= \left(\frac{1}{\sqrt{2\pi}\sigma}\right)^N e^{\frac{-\sum_i x_i^2}{2\sigma^2}} \cdot e^{\frac{-N\mu^2}{2\sigma^2}+\frac{N\mu\overline{\mu}_{ML}}{\sigma^2}}\end{aligned} \tag{2.27}$$

entrambi gli stimatori $\overline{\sigma}_{ML}$ e $\overline{\mu}_{ML}$ sono sufficienti. Il metodo ML ci permette dunque di estrarre dal campione tutta l'informazione disponibile su θ.

2.5 Stimatori corretti di varianza minima

Il **teorema di Cramèr–Rao**[3] stabilisce che, dato un campione, esiste un valore minimo per la varianza di qualunque stimatore di una variabile θ:

$$\sigma^2(\overline{\theta}) \geq \frac{\left(\frac{\partial E(\overline{\theta})}{\partial \theta}\right)^2}{I_\theta} \tag{2.28}$$

dove

$$I_\theta = E_{\mathcal{S}}\left[\left(\frac{\partial \log \mathcal{L}(\mathcal{S},\theta)}{\partial \theta}\right)^2\right] = -E_{\mathcal{S}}\left(\frac{\partial^2 \log \mathcal{L}(\mathcal{S},\theta)}{\partial \theta^2}\right) \tag{2.29}$$

è chiamata **informazione di Fisher** ($E_{\mathcal{S}}$ rappresenta il valore atteso nello spazio dei campioni). Per uno stimatore corretto, è l'inverso della varianza minima, e quindi quantifica l'informazione contenuta in $\mathcal{S}$per poter stimare θ.
Uno stimatore di varianza minima è detto **efficiente** (o "il più efficiente"). Si dimostra che gli stimatori efficienti sono anche sufficienti.

Questo importante teorema ci permette quindi di stabilire se la stima di un parametro possa essere migliorata o se abbiamo usato in modo ottimale tutta l'informazione disponibile nei nostri dati. Uno stimatore corretto ed efficiente è normalmente il migliore possibile, a meno che non vogliamo che sia anche robusto.
Come esempio, consideriamo ancora il caso della stima del valore atteso μ da un campione normale, con likelihood data dalla 2.20. L'informazione di Fisher sarà

$$I_\mu = -E_{\mathcal{S}}\left(\frac{\partial^2 \log \mathcal{L}(\mathcal{S},\mu)}{\partial \mu^2}\right) = \frac{N}{\sigma^2} \tag{2.30}$$

che è proprio l'inverso della varianza della media aritmetica. Nel caso gaussiano la media, che è lo stimatore ML del valore atteso, è dunque efficiente.
Notiamo che nell'assunzione di misure indipendenti

$$\log \mathcal{L} = \sum_{i=1}^{N} \log \mathcal{d}(\underline{x}_i) \quad \Longrightarrow \quad I_\theta \propto N$$

e dunque l'errore di uno stimatore efficiente è sempre proporzionale a $1/\sqrt{N}$.

Il teorema può essere esteso al caso di n parametri $\underline{\theta}$. In tal caso si definisce una matrice simmetrica, detta matrice di informazione di Fisher, come

$$F_{ij} = E_{\mathcal{S}}\left(-\frac{\partial^2 \log \mathcal{L}(\mathcal{S},\underline{\theta})}{\partial \theta_i \partial \theta_j}\right) \tag{2.31}$$

e il limite di Cramèr–Rao si applica alla matrice varianza–covarianza degli stimatori V_θ:

$$(V_\theta)_{ij} \geq (F^{-1})_{ij} \tag{2.32}$$

[3]si veda [1] per la dimostrazione. Il teorema vale se la funzione densità delle variabili $\underline{x}$è derivabile in θ e definita in un intervallo di valori i cui estremi non dipendono da θ.

Si ha dunque un limite anche per le covarianze. Nel caso della gaussiana, abbiamo visto che $\partial^2 \log \mathcal{L}/\partial\mu\partial\sigma = 0$, per cui gli stimatori efficienti di μ e σ sono indipendenti.

2.6 Limite degli stimatori ML

Sviluppiamo la funzione di likelihood nell'intorno del massimo:

$$\log \mathcal{L}(\theta) = \log \mathcal{L}(\overline{\theta}_{ML}) + \frac{\partial^2 \log \mathcal{L}}{\partial \theta^2}(\overline{\theta}_{ML}) \cdot \frac{(\theta - \overline{\theta}_{ML})^2}{2} + \dots \tag{2.33}$$

All'ordine più basso, il profilo del logaritmo di $\mathcal{L}$ intorno al massimo è parabolico, e ci aspettiamo dunque che $\mathcal{L}$ sia approssimativamente una gaussiana

$$\mathcal{L}(\theta) \simeq \mathcal{L}(\overline{\theta}_{ML}) \exp\left(\frac{-(\theta - \overline{\theta}_{ML})^2}{2s^2}\right) \tag{2.34}$$

dove

$$s^2 = -\frac{1}{\frac{\partial^2 \log \mathcal{L}}{\partial \theta^2}(\overline{\theta}_{ML})} \tag{2.35}$$

Si dimostra (vedi [1]), come conseguenza del teorema del limite centrale, che nel limite di infinite misure i termini di ordine superiore nell'eq. 2.33 tendono a zero, e s^2 tende, per la legge dei grandi numeri, alla media nello spazio dei campioni dell'espressione 2.35, ovvero all'inverso dell'informazione di Fisher. Si ha dunque

$$lim_{N\to\infty}\ \mathcal{L}(\theta) = \mathcal{L}(\overline{\theta}_{ML}) \exp\left(\frac{-(\theta - \overline{\theta}_{ML})^2 I_\theta}{2}\right) \tag{2.36}$$

In questo limite la PDF di $\overline{\theta}_{ML}$ diventa pure una gaussiana con media θ e varianza $1/I_\theta$. Gli stimatori ML sono dunque, nel limite di infinite misure, corretti ed efficienti. Se questo è rigorosamente vero nel limite asintotico, può essere vero con buona approssimazione anche con un numero finito di misure.

Nel caso frequente in cui l'espressione della likelihood non sia calcolabile analiticamente, il massimo della funzione $\mathcal{L}$, o più comunemente il minimo di $-\log \mathcal{L}$, viene calcolato numericamente. Il profilo della funzione ci può dire se siamo ragionevolmente vicini al limite: se $-\log \mathcal{L}$ è descritto da una parabola in un intorno di alcune σ intorno al minimo, possiamo assumere che lo stimatore sia, con buona approssimazione, corretto ed efficiente. Il profilo ci permette anche di stimare l'errore sulla stima $\sigma(\overline{\theta}_{ML})$: nel limite, i valori di θ per cui $\log \mathcal{L} = \log \mathcal{L}(\overline{\theta}_{ML}) - 1/2$ corrispondono a $\overline{\theta}_{ML} \pm \sigma(\overline{\theta}_{ML})$.

Esempio 2.6.1 *Vita media del Polonio*
Supponiamo di voler misurare la vita media dell'isotopo Po_{214} osservando la catena di decadimenti nucleari

$$Bi_{214} \to Po_{214} \to Pb_{210}$$

Il nostro apparato misura la differenza di tempo t_{exp} fra i due decadimenti con risoluzione $\sigma_t = 0.07$ ms, ma ci fornisce solo valori maggiori di 0. I dati ottenuti si trovano nel file /afs/math.unifi.it/service/Rdsets/polonium

La distribuzione della variabile $t_{exp} = t + \epsilon$ è la convoluzione della distribuzione esponenziale del decadimento e della distribuzione gaussiana che descrive l'errore ϵ, che abbiamo già implementato nell'esempio 1.9.2 e chiamato *expconv()* . In questo caso bisogna tener conto che la distribuzione è definita solo per $t_{exp} > 0$. E' dunque necessario rinormalizzare la funzione in modo cha l'integrale sia 1:

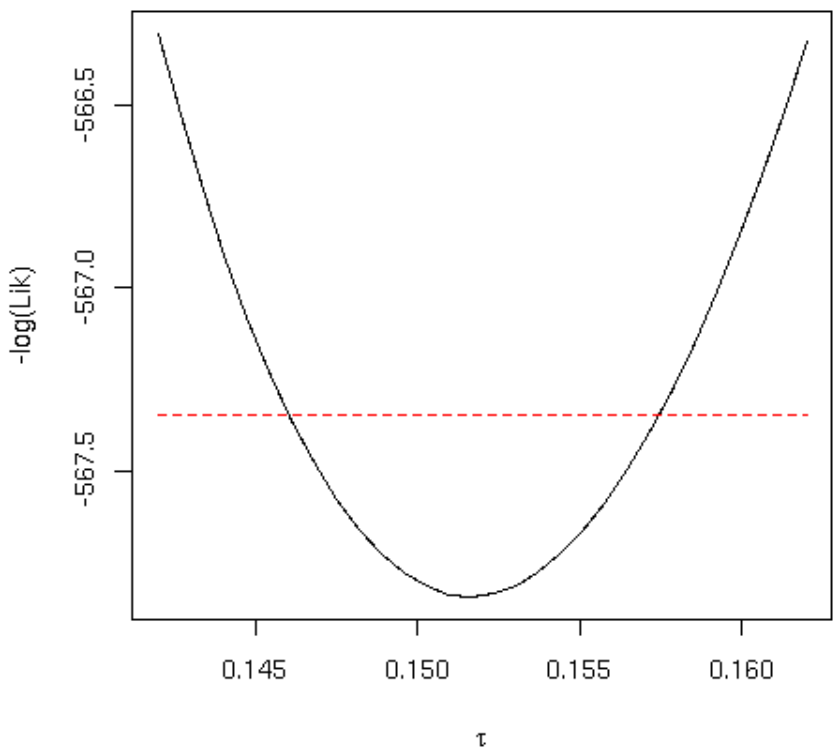
```
sample = scan("/afs/math.unifi.it/service/Rdsets/polonium")
sigmaT=0.07
sexpconv = function(t,sigma=sigmaT,tau=1) {
 (t>0) * expconv(t,sigma,tau) /
    integrate(expconv,lower=0,upper=20*tau,sigma=sigma,tau=tau)$value
}
```

Il valore atteso di questa distribuzione non è, a differenza di *expconv()* , pari alla vita media τ. Ricaviamo allora τ tramite il principio di ML, minimizzando numericamente la funzione $-log(\mathcal{L})$:

```
expconvlik = function(tau,sample) {
  out=c()
  for (i in 1:length(tau) ) {
   out[i] = -sum( log(sexpconv(sample,sigma=sigmaT,tau=tau[i])) )
  }
  out
}
curve(expconvlik(x,sample=sample),0.142,0.162,n=30,
           xlab=expression(tau), ylab="-log(Lik)")
min=expconvlik(0.1516,sample=sample)
lines(c(0.142,0.162),c(min+0.5,min+0.5),lty=2,col="red")
```

Possiamo stimare dal grafico

$$\tau = 0.152 \pm 0.006$$



2.6.1 Caso multivariato

Nel caso si un'analisi in cui si vogliano stimare più parametri, rappresentati dal vettore $\underline{\theta}$, l'eq. 2.33 diventa

$$\log \mathcal{L}(\underline{\theta}) = \log \mathcal{L}(\overline{\underline{\theta}}_{ML}) + \frac{1}{2}(\underline{\theta} - \overline{\underline{\theta}}_{ML})^T H(\underline{\theta} - \overline{\underline{\theta}}_{ML}) + \dots \tag{2.37}$$

dove H è la matrice hessiana di $\log \mathcal{L}$ calcolata nel massimo.
Nel limite di infinite misure

$$lim_{N\to\infty}\ \mathcal{L}(\underline{\theta}) = \mathcal{L}(\overline{\underline{\theta}}_{ML}) \exp\left(-\frac{1}{2}(\underline{\theta} - \overline{\underline{\theta}}_{ML})^T F(\underline{\theta} - \overline{\underline{\theta}}_{ML})\right) \tag{2.38}$$

dove F è la matrice di informazione di Fisher 2.31. La PDF di $\overline{\underline{\theta}}_{ML}$ diventa in tal caso una gaussiana multivariata la cui matrice varianza–covarianza è data dall'inverso di F.

Esempio 2.6.2 *Estrazione di un segnale con fondo uniforme*
I dati nel file
/afs/math.unifi.it/service/Rdsets/gaussconfondo.rdata
rappresentano misure di energia di radiazioni γ provenienti da un decadimento nucleare caratterizzato da un'energia di circa 5 MeV e compresi nell'intervallo di energia fra 0 e 10 MeV.
Ci si aspetta che tutti i fotoni osservati abbiano la stessa energia μ, ma il nostro rivelatore ha una risoluzione di 0.3 MeV (l'errore della misura segue una distribuzione gaussiana con valore

atteso nullo e deviazione standard $\sigma = 0.3$ MeV).
Sono inoltre presenti eventi di fondo (ovvero, segnali non dovuti alla sorgente osservata) che supponiamo avere una distribuzione di probabilità uniforme.
Vogliamo ricavare la miglior stima dell'energia μ e della frazione di eventi di fondo α nel campione.

Parametrizziamo la distribuzione di probabilità della variabile misurata, l'energia x:

$$f(x;\mu,\alpha) = (1-\alpha)\phi_{Gauss}(x;\mu,\sigma_G) + \alpha d_{unif}(x;x_{min},x_{max})$$

dove $\sigma_G = 0.3$ MeV, $x_{min} = 0$, $x_{min} = 10$ MeV.

Una prima stima dei parametri μ e α può essere ottenuta stimando il valore atteso e la varianza della distribuzione:

$$\begin{aligned} E(x) &= \int x f(x) dx = (1-\alpha)\int x\phi_{Gauss}(x)dx + \alpha\int x d_{unif}(x)dx = \\ &= (1-\alpha)\mu + \alpha E_u \end{aligned}$$

$$\begin{aligned} \sigma^2(x) &= E(x^2) - E(x)^2 = (1-\alpha)\int x^2\phi_{Gauss}(x)dx + \alpha\int x^2 d_{unif}(x)dx - E(x)^2 = \\ &= (1-\alpha)(\sigma_G^2+\mu^2) + \alpha(\sigma_u^2+E_u^2) - ((1-\alpha)\mu+\alpha E_u)^2 = \\ &= (1-\alpha)\sigma_G^2 + \alpha\sigma_u^2 + \alpha(1-\alpha)(\mu-E_u)^2 \simeq (1-\alpha)\sigma_G^2 + \alpha\sigma_u^2 \end{aligned}$$

dove $E_u = 5MeV$ e $\sigma_u^2 = 100/12$ MeV^2 sono il valore atteso e la varianza della distribuzione uniforme, e si è ottenuta l'ultima relazione assumendo $|\mu - E_u|^2 << \sigma_u^2$.

Ricaviamo dunque gli stimatori

$$\bar{\alpha} = \frac{\overline{\sigma^2(x)} - \sigma_G^2}{\sigma_u^2 - \sigma_G^2}$$

$$\bar{\mu} = \frac{\overline{E(x)} - E_u \cdot \bar{\alpha}}{1-\bar{\alpha}}$$

usando i noti stimatori corretti di $E(x)$ e $\sigma^2(x)$

$$\bar{x} = \overline{E(x)} = \sum_i x_i/N$$

$$\overline{\sigma^2(x)} = \sum_i (x_i - \bar{x})^2/(N-1)$$

Gli errori standard su queste stime saranno:

$$\sigma(\bar{\alpha}) = \frac{\overline{\sigma_G(\overline{\sigma^2(x)})}}{\sigma_u^2 - \sigma_G^2}$$

$$\sigma(\bar{\mu}) \simeq \frac{\overline{\sigma_G(\overline{E(x)})}}{1-\bar{\alpha}} = \frac{1}{1-\bar{\alpha}}\sqrt{\frac{\overline{\sigma^2(x)}}{N}}$$

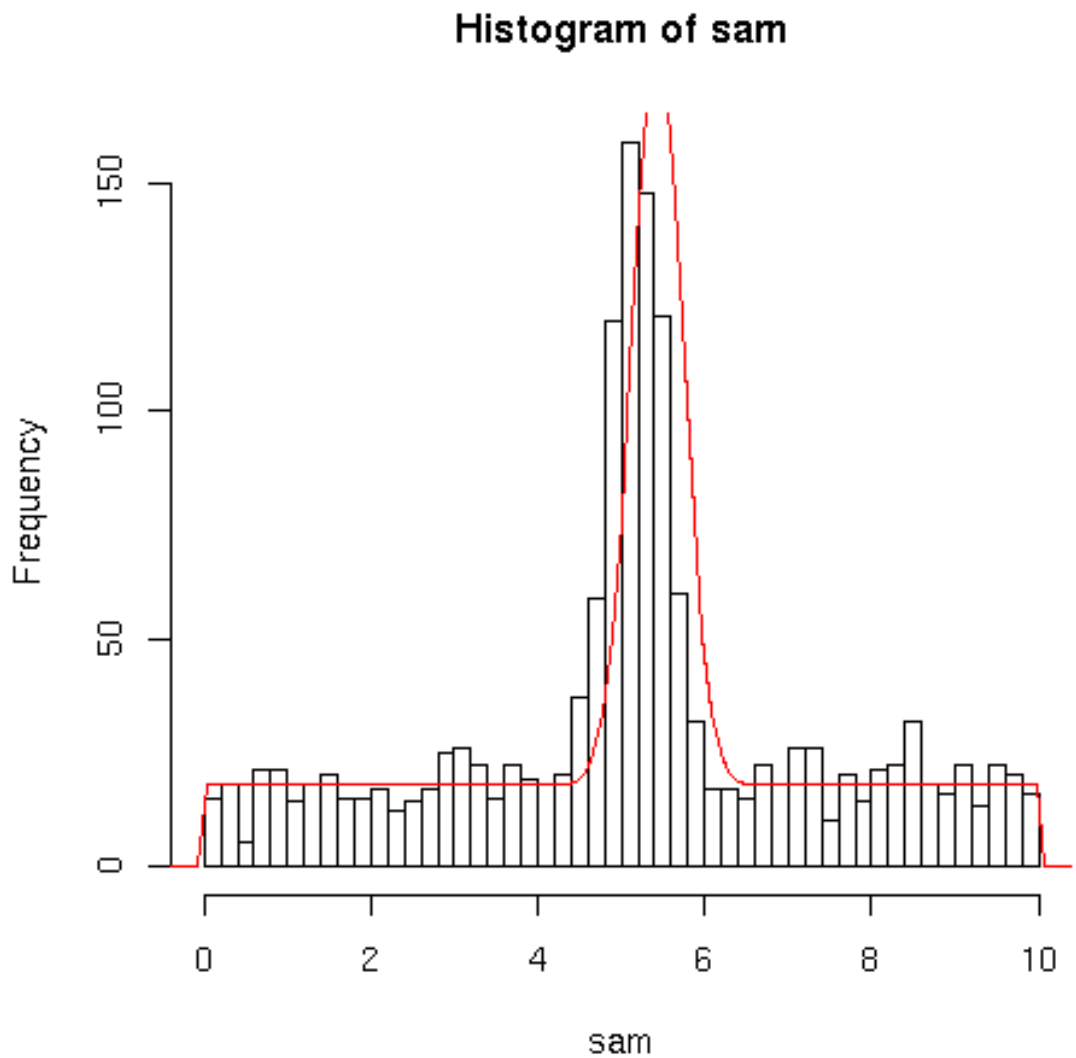
dove abbiamo trascurato l'incertezza su $\bar{\alpha}$ nell'espressione di $\sigma(\bar{\mu})$ e

$$\overline{\sigma_G(\overline{\sigma^2(x)})} = \sqrt{\frac{1}{N}\left(\mu_4(x) - \frac{N-3}{N-1}\mu_2^2(x)\right)} \simeq \sqrt{\frac{\overline{\mu_4(x)} - \overline{\sigma^2(x)}^2}{N}}$$

$$\overline{\mu_4(x)} = \frac{\sum_i (x_i - \bar{x})^4}{N-1}$$

Stimiamo i valori numerici e verifichiamo graficamente il risultato:

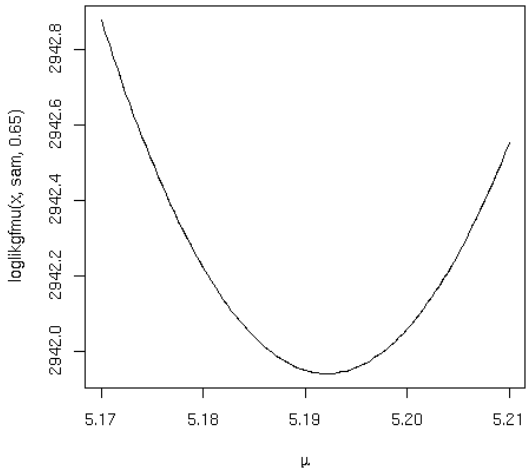
```
gaussconfondo= function(x,alpha,mu,s1=0.3) {
 (1-alpha)*dnorm(x,mean=mu,sd=s1)+alpha*dunif(x,min=0,max=10)
}

> sam=scan("gaussconfondo.rdata")
> n=length(sam)
> s1=0.3
> myalpha=(var(sam)-s1^2)/(25/3-s1^2)
> mu4=mean((sam-mean(sam))^4)
> dvar=sqrt((mu4-var(sam)^2)/n)
> mydalpha=dvar/(25/3-s1^2)

> mymu=(mean(sam)-5*myalpha)/(1-myalpha)
> mydmu = sqrt(var(sam)/(n*(1-myalpha)^2))

> cat ("stima di alpha: ",myalpha," +/- ",mydalpha,"\n")
stima di alpha:  0.59  +/-  0.02
> cat ("stima di mu: ",mymu," +/- ",mydmu,"\n")
stima di mu:  5.43  +/-  0.14

hist(sam,breaks=seq(0.,10.,0.2))
curve(gaussconfondo(x,alpha=myalpha,mu=mymu)*n*0.2,add=T,col="red")
```

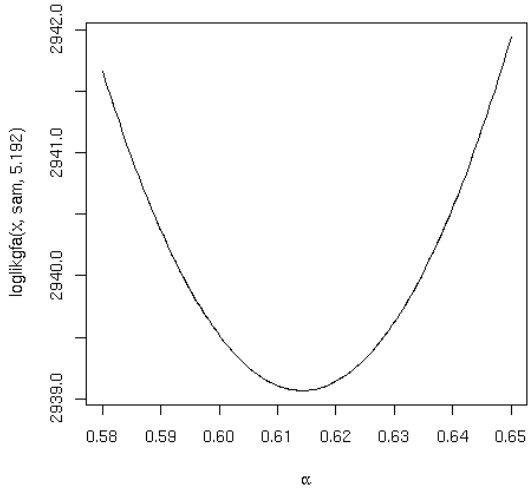


E' evidente che la stima di μ, seppur sensata entro l'errore stimato, può essere migliorata. Questo non è sorprendente dal momento che, usando la media aritmetica, abbiamo utilizzato con lo stesso peso tutti gli eventi, anche quelli del fondo che non portano alcuna informazione su μ ma aggiungono solo "rumore" alla nostra stima. Il metodo di ML ci fornisce invece uno stimatore che estrae dai dati il massimo dell'informazione. Ci aspettiamo inoltre che, vista la statistica relativamente elevata del campione, lo stimatore abbia alta efficienza e bias trascurabile.
Calcoliamo il massimo della funzione di likelihood numericamente minimizzando la quantità $-\log \mathcal{L}$. In generale, dovremmo trovare il minimo variando simultaneamente i due parametri μ e α. Tuttavia ci aspettiamo che la correlazione fra i due parametri sia piccola, poichè i parametri descrivono caratteristiche diverse del campione. E' dunque ragionevole stimare un parametro alla volta, ed eventualmente raffinare la stima procedendo in modo iterativo:

```
loglikgfa = function(a,sample,m) {
 out=c()
 for (i in 1:length(a)) {
  out[i] = -sum(log(gaussconfondo(sample,alpha=a[i],mu=m)))
 }
```

```
 out
}

loglikgfmu = function(m,sample,a) {
 out=c()
 for (i in 1:length(m)) {
  out[i] = -sum(log(gaussconfondo(sample,alpha=a,mu=m[i])))
 }
 out
}

# prima stima di mu e alpha
curve(loglikgfmu(x,sam,myalpha),5.15,5.25)
# possiamo stimare visivamente dal grafico
# mu_ML = 5.192 +/- 0.016

curve(loglikgfa(x,sam,mymu),0.5,0.7)
# possiamo stimare visivamente dal grafico
# alpha_ML = 0.650 +/- 0.015

# seconda iterazione
curve(loglikgfmu(x,sam,0.650),5.17,5.21,xlab=expression(mu))
curve(loglikgfa(x,sam,5.192),0.55,0.65,xlab=expression(alpha))
```





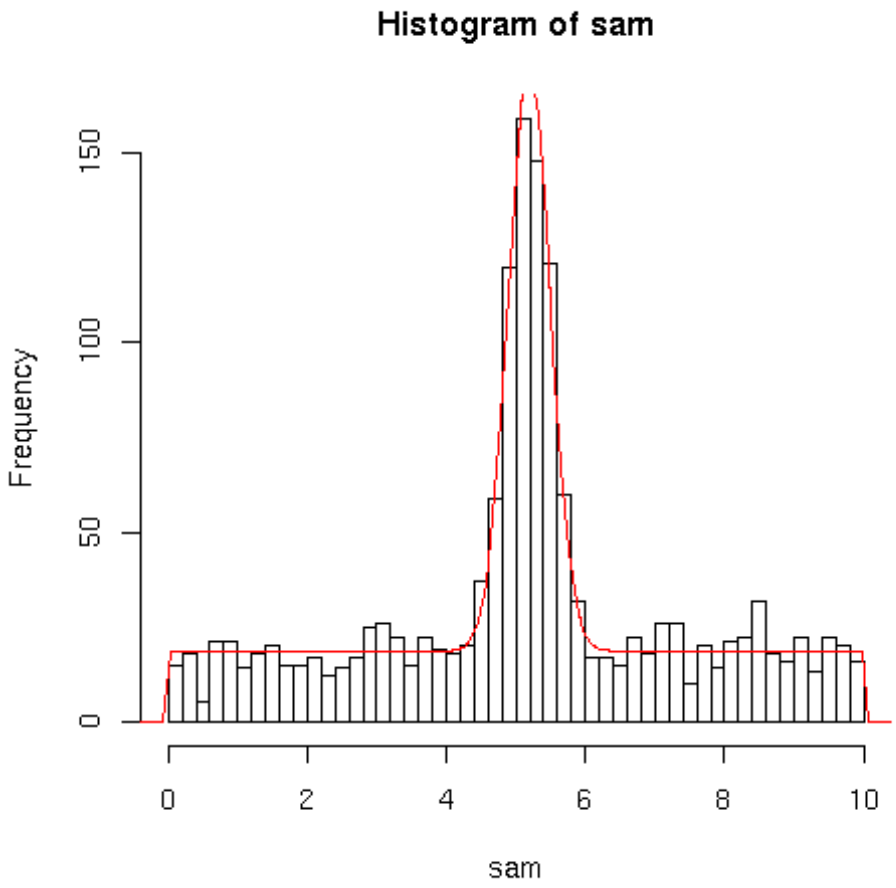
Dai grafici possiamo stimare, valutando i valori per cui $-\log L$ differisce di $1/2$ dal minimo:

$$\bar{\mu}_{ML} = 5.192 \pm 0.016$$

$$\bar{\alpha}_{ML} = 0.614 \pm 0.015$$

Una ulteriore iterazione non produce variazioni rilevanti. Come ci si poteva aspettare, la precisione della nostra stima è decisamente migliorata, in particolare per μ. Notiamo anche come il profilo di $\log L$ sia parabolico, confermandoci che siamo prossimi al limite asintotico in cui gli stimatori di ML sono corretti ed efficienti.
Verifichiamo infine graficamente l'accordo fra i dati e la miglior stima ottenuta:

```
hist(sam,breaks=seq(0.,10.,0.2))
curve(gaussconfondo(x,alpha=0.614,mu=5.192)*n*0.2,add=T,col="red")
```



2.7 Simulazione di campioni: tecniche Montecarlo

Come già visto in numerosi esempi, le distribuzioni di probabilità di quantità di interesse, in problemi complessi non risolvibili analiticamente, possono essere stimate numericamente tramite la simulazione al computer di un campione di valori estratto "a caso" secondo il modello probabilistico ipotizzato, una procedura chiamata in gergo "simulazione MonteCarlo". La generazione di numeri casuali al computer è una applicazione di primaria importanza in matematica (si pensi ad esempio alla crittografia) e, lungi dal voler dare qui un quadro esaustivo della materia, accenneremo soltanto in questo paragrafo ad alcune tecniche di simulazione elementari.

Il primo passo consiste nell'avere un algoritmo che generi una sequenza di valori "casuali", tipicamente distribuiti uniformemente in un intervallo dato. Qualunque algoritmo matematico darà, a partire da un valore di partenza (detto "seme"), una sequenza predicibile, anche se i valori seguono apparentemente la distribuzione voluta, e si parla in tal caso di numeri **pseudo–random**. Uno dei più semplici esempi di algoritmi di generazione pseudo–random è il seguente:

dato il seme n_0, si costruiscono i valori successivi come

$$n_{i+1} = mod\left(\frac{an_i}{m}\right) \tag{2.39}$$

dove a e m sono numeri interi maggiori di 1. Si ottiene così una sequenza di numeri con valore discreto fra 1 e m-1. Con m sufficientemente grande, i valori $(n_i - 1)/(m - 2)$ seguono approssimativamente la distribuzione uniforme fra 0 e 1.
L'algoritmo può essere implementato in R dal seguente codice:

```
myrandom= function(N, seed=2501, m=500000, a=3456 ) {
 out=c(seed)
 i=1
 while (i < N) {
  out[i+1]= (a*out[i]) %% m
  i=i+1
 }
 (out-1)/(m-2)
}

hist( myrandom(10000), breaks=50)
```

Il limite di un algoritmo pseudo–random è la lunghezza necessariamente finita della sequenza di numeri che può essere prodotta senza ripetersi . Questa lunghezza e l'efficienza del calcolo (rapidità dell'algoritmo) sono i due parametri che caratterizzano le performances di un algoritmo di generazione, e che sono ovviamente in contrasto fra loro. La scelta dell' algoritmo più opportuno dipende dunque dall'applicazione ed è un compromesso fra le due richieste.
Nel software R, sono implementati diversi algoritmi di generazione, si veda
help(.Random.seed)
per maggiori dettagli. E' possibile definire il seme tramite il comando
set.seed(x)
in modo da poter riprodurre più volte la stessa sequenza di numeri pseudo–random, cosa che può risultare utile nel confrontare due analisi sullo stesso campione MonteCarlo.

Esistono anche generatori "true–random" (o misti), in cui i valori sono ottenuti da un processo fisico intrinsecamente casuale, come il rumore termico o un fenomeno quantistico come l'effetto fotoelettrico. Ne è un esempio il generatore implementato nello pseudo–file */dev/random* del sistema operativo Linux, che sfrutta le fluttuazioni sui tempi di arrivo dei segnali da periferiche quali il mouse e la tastiera.
Una sequenza di numeri random da questo generatore può essere ottenuta tramite il comando

```
od /dev/random
```

Si noterà che la sequenza si interrompe in assenza di un input sufficientemente casuale, ma è sufficiente muovere un pò il mouse per vederla ripartire...

Il passo successivo consiste nell'utilizzare la sequenza di valori (pseudo–)random per simulare una generica distribuzione densità $f(x)$. Se r è una variabile aleatoria distribuita uniformemente fra 0 e 1, dovremo applicare il cambio di variabile $x(r)$ tale che

$$f(x)dx = d_u(r)dr \tag{2.40}$$

ovvero

$$p(x) = \int_{-\infty}^{x} f(x')dx' = \int_{0}^{r} d_u(r)dr = r \tag{2.41}$$

La soluzione è dunque semplicemente la funzione quantile

$$x(r) = q(r) \tag{2.42}$$

Ad esempio, da un sequenza r_i si può ottenere un campione distribuito secondo la distribuzione esponenziale con parametro τ come $x_i = -\tau log(1 - r_i)$

Nel software R, ogni funzione di probabilità implementata ha il suo generatore, ad esempio rnorm(), rexp(), rpois(), etc. Questi sono equivalenti ad applicare la corrispondente funzione quantile (qnorm(), qexp(), ...) ad una sequenza ottenuta con runif().

Esempio 2.7.1 *Evoluzione di un prezzo*
Si vuole simulare l'evoluzione futura del prezzo di un prodotto, assumendo che l'aumento annuale segua il semplice modello

$$P(y + 1) = P(y) \cdot r \cdot (1 + s)$$

dove $P(y)$ indica il prezzo per l'anno y, $r = 1.02$ è il rate di inflazione medio e lo spread s è una variabile aleatoria che segue una distribuzione gaussiana con valore atteso nullo e deviazione standard $\sigma = 0.3$. Sapendo che il prezzo corrente è pari a 1€, ci interessa la probabilità che il prezzo superi 3€ fra 5 anni.

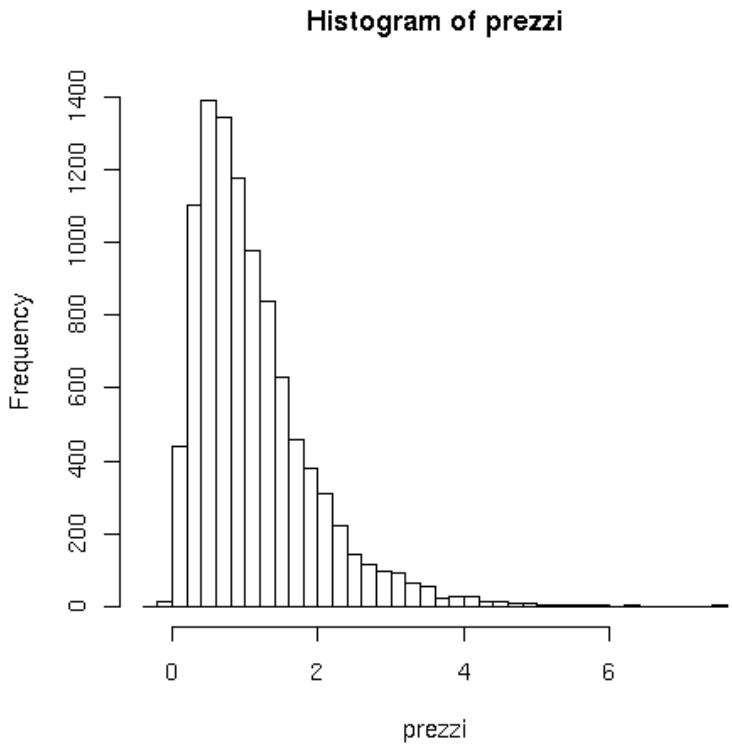
La variabile prezzo fra n anni è data da

$$P(y + n) = P(y) \cdot r^n \cdot \prod_{i=1}^{n}(1 + s_i)$$

La sua funzione densità è dunque ottenibile dalla convoluzione di Mellin di n gaussiane, problema risolvibile tramite le funzioni di Bessel. Più semplicemente, possiamo risolvere il problema con una simulazione MonteCarlo:

```
prezzo.corrente=1
rinfl= 1.02
spread=0.3
anno.corrente=2007

prezzofuturo = function(anno) {
  n=anno-anno.corrente
  s = rnorm(n,sd=spread)
  prezzo.corrente * rinfl^n * prod(1+s)
```

```
}

simprezzo= function(anno,N=10000) {
 prezzi = c()
 for (i in 1:N) {
  prezzi[i] = prezzofuturo(anno)
 }
 hist(prezzi,breaks=50)
 cat("stima del valore atteso : ",mean(prezzi)," +/- ",sqrt(var(prezzi)/N)," \n");
 cat("stima della deviazione standard : ",sd(prezzi)," +/- ",
sqrt((sum((prezzi-mean(prezzi))^4)/(N-1)-var(prezzi)^2)/N)," \n");
# calcoliamo la frazione oltre 3 e il suo errore
 piuditre = length(prezzi[prezzi > 3])/N
 dpiuditre = sqrt(piuditre*(1-piuditre)/N)
 cat("frazione oltre 3 euro: ",piuditre," +/- ",dpiuditre," \n");
}
> simprezzo(2012)
stima del valore atteso :  1.114  +/-  0.008
stima della deviazione standard :  0.818  +/-  0.017
frazione oltre 3 euro:  0.0335  +/-  0.0018
```



Si noti che il valore atteso e la deviazione standard avrebbero potuto essere stimati usando la formula di propagazione degli errori:

$$E(P(y+n)) = P(y) \cdot r^n E(\prod_{i=1}^{n}(1+s_i)) = P(y) \cdot r^n \prod_{i=1}^{n} E(1+s_i) = P(y) \cdot r^n$$

$$\sigma((P(y+n)))/E(P(y+n)) \simeq \sqrt{\sum_{i=1}^{n} (\sigma(1+s_i)/E(1+s_i))^2} = \sqrt{n}\ \sigma$$

Per $n = 5$ otteniamo dunque $E(P(y+n)) = 1.104$, $\sigma((P(y+n))) = 0.74$. L'approssimazione lineare usata nella propagazione degli errori ci avrebbe portato ad una sottostima della deviazione standard. In questa approssimazione, avremmo assunto la distribuzione di $P(Y+n)$ gaussiana e calcolato la probabilità di superare 3 euro come

```
> atteso = prezzo.corrente * rinfl^5
> gsigma = atteso * spread * sqrt(5)
> 1 - pnorm((3-atteso)/gsigma)
[1] 0.005236028
```

ottenendo 0.5% anzichè il valore corretto $(3.3 \pm 0.2)\%$ ottenuto dalla simulazione.

2.7.1 Metodo di von Neumann

Se la funzione quantile non è ricavabile analiticamente, il problema può essere risolto tramite un algoritmo numerico. L'algoritmo più semplice, dovuto a von Neumann, consiste nell'estrarre coppie di punti (x, y), distribuite uniformemente negli intervalli $a < x < b$, $0 < y < max_{[a,b]}(f(x))$. I valori di x che soddisfano la relazione $y \leq f(x)$ saranno distribuiti secondo la PDF $f(x)$ nell'intervallo $[a, b]$.

Esempio 2.7.2 *Un campione ottenuto col metodo di Von Neumann*
Si vuole simulare un campione di valori secondo la distribuzione

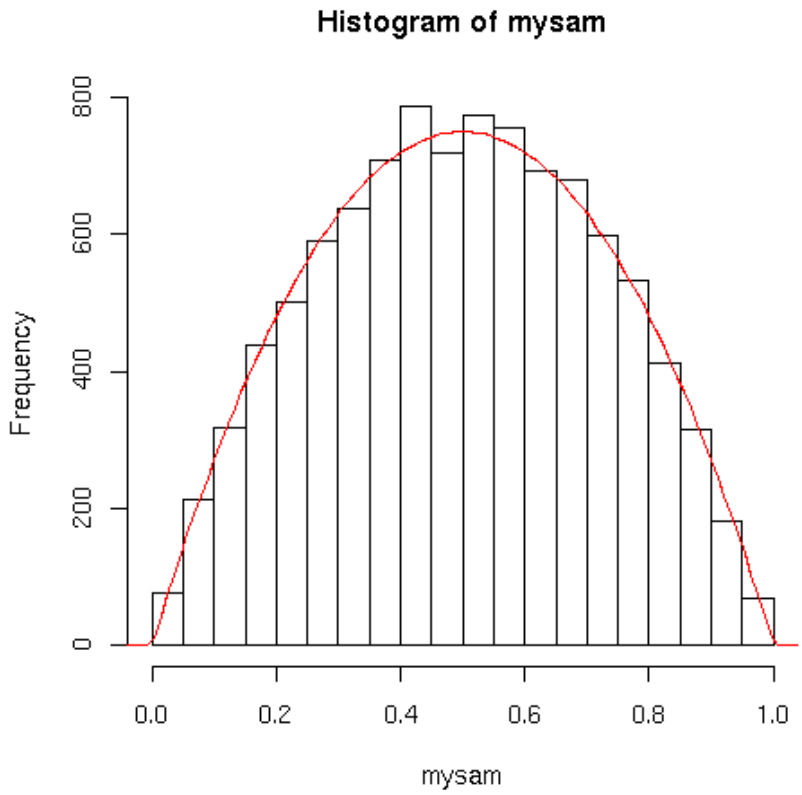
$$f(x) = 6x(1-x)$$

definita fra 0 e 1.

Scriviamo la funzione densità e il generatore random usando il metodo *accept/reject* di Von Neumann: generiamo coppie di punti (X, Y) distribuite uniformemente negli intervalli $0 < X < 1$, $0 < Y < max(f) = 1.5$, e accettiamo solo i punti tali che $Y \leq f(X)$

```
dpol = function(x) {
 ifelse(x<0 | x>1 , 0,6*x*(1-x) )
}

rpol = function(N) {
 out =c()
 for (i in 1:N) {
  y = runif(1,max=1.5)
  r = runif(1)
  while ( y > dpol(r) ) {
   y = runif(1,max=1.5)
   r = runif(1)
  }
```

```
  out[i]=r
 }
 out
}

# metodo alternativo (non garantisce la lunghezza del vettore prodotto):
rpol2 = function(N) {
 x=runif(N)
 y=runif(N,max=1.5)
 x[y <= dpol(x)]
}

mysam=rpol(10000)
hist(mysam,breaks=seq(0,1,0.05))
curve(dpol(x)*0.05*10000,add=T,col="red")
```



2.8 Intervalli di confidenza

Per quantificare l'incertezza sulle nostre stime, abbiamo finora utilizzato la stima della deviazione standard dello stimatore, spesso chiamato **"errore standard"** della stima. La giustificazione è intuitiva: se, dato θ, $\overline{\theta}$ ha valore atteso θ e deviazione standard $\sigma(\overline{\theta})$, allora dobbiamo aspettarci che la differenza fra θ e il valore osservato $\overline{\theta}_{obs}$ sia dell'ordine di $\sigma(\overline{\theta})$. Vogliamo ora dare

un significato più preciso, in termini di probabilità, a questo errore.
Cominciamo col notare che vi sono casi in cui la stima di $\sigma(\overline{\theta})$ può essere grossolanamente sbagliata. Consideriamo ad esempio il caso di un sondaggio elettorale in cui stimiamo la percentuale attesa per un dato partito p dalla frequenza $\overline{p} = N_g/N$ di persone intervistate che dichiarano di votarlo. Ci aspettiamo (cfr. par. 1.11) che lo stimatore $\overline{p}$ sia una variabile aleatoria con deviazione standard pari a $\sqrt{\frac{p(1-p)}{N}}$, e frequentemente si valuta dunque l'errore della stima come

$$\overline{\sigma(\overline{p})} \simeq \sqrt{\frac{\overline{p}(1-\overline{p})}{N}} \tag{2.43}$$

Applicando questa formula al caso $N_g = N = 5$, si ottiene $\overline{p} = 1$ con errore nullo, un risultato chiaramente assurdo che discende dal fatto di aver preso come valore di p il valore stimato $\overline{p}$, senza considerare gli altri valori di p compatibili col risultato osservato dello stimatore.

Per dare un preciso significato statistico alla nostra stima, dobbiamo quindi considerare la probabilità di ottenere $\overline{\theta}_{obs}$ per tutti i valori possibili della grandezza p e definire un criterio per quantificare in termini probabilistici la nostra confidenza sul risultato della stima. Il problema non è privo di sottigliezza: nell'approccio frequentista non ha senso associare una probabilità al valore del parametro θ che vogliamo stimare, non essendo questo una variabile aleatoria. Il **livello di confidenza** α sul risultato di una stima deve essere invece definito come la probabilità che il risultato – solitamente un intervallo di valori – sia giusto, ovvero che il valore vero sia contenuto nell'intervallo. In altre parole, non ha senso dire "la probabilità che la massa dell'elettrone abbia un certo valore", ma ha senso chiedersi qual è la probabilità che il risultato di una misura di massa dell'elettrone contenga il valore vero.

La procedura per ottenere gli **intervalli di confidenza** così definiti, dovuta a Neyman[10], consiste nel costruire una "banda di confidenza" nel piano $(\theta, \overline{\theta})$ contenente l'intervallo dei valori di $\overline{\theta}$ che ci si aspetta di ottenere con probabilità α per ciascun valore di θ:

$$\int_{v(\theta)}^{u(\theta)} \mathcal{L}(\overline{\theta}|\theta) d\overline{\theta} = \alpha \tag{2.44}$$

come illustrato in figura 2.1.
Solitamente si considera un intervallo di confidenza centrale, per cui

$$\int_{-\infty}^{v(\theta)} \mathcal{L}(\overline{\theta}|\theta) d\overline{\theta} = \int_{u(\theta)}^{\infty} \mathcal{L}(\overline{\theta}|\theta) d\overline{\theta} = (1-\alpha)/2 \tag{2.45}$$

e la banda di confidenza è definita dunque dai quantili

$$\begin{aligned} v(\theta) &= q_{\overline{\theta}}\left(\frac{1-\alpha}{2}\middle|\theta\right) \\ u(\theta) &= q_{\overline{\theta}}\left(1-\frac{1-\alpha}{2}\middle|\theta\right) \end{aligned} \tag{2.46}$$

L'intervallo di confidenza $[a, b]$ è ottenuto semplicemente intersecando la banda di confidenza con il valore osservato dello stimatore $\overline{\theta}_{obs}$:

$$\begin{aligned} \overline{\theta}_{obs} &= u(a) \\ \overline{\theta}_{obs} &= v(b) \end{aligned} \tag{2.47}$$

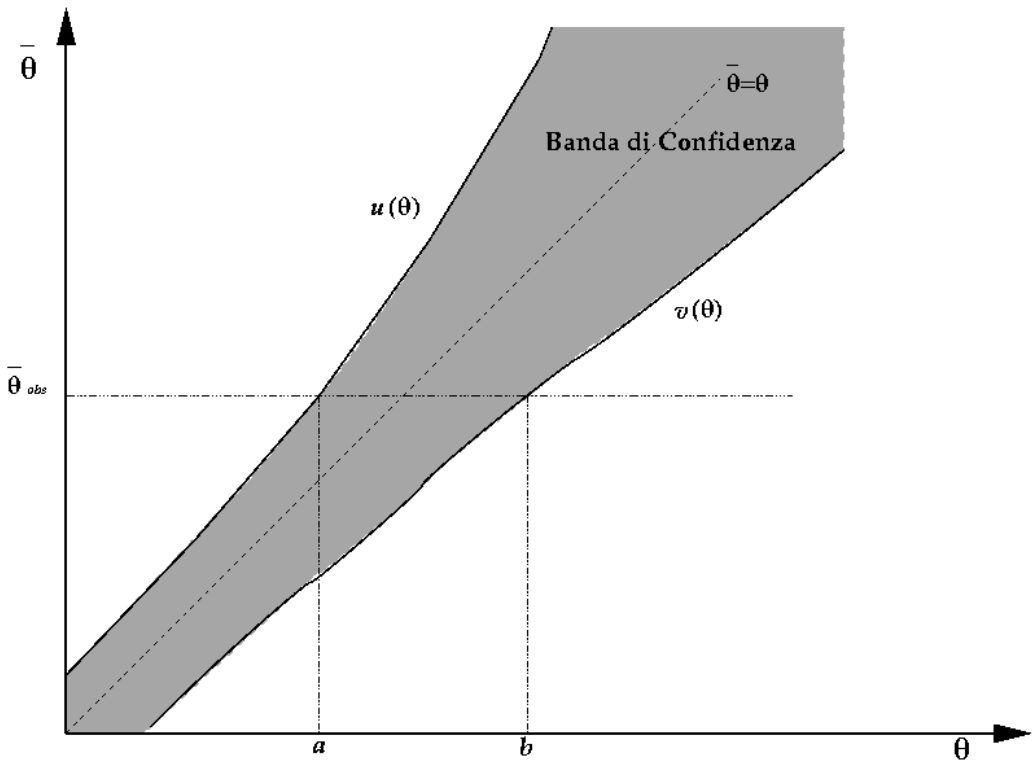


Figura 2.1: Costruzione dell'intervallo di confidenza col metodo di Neyman.

Infatti, l'intervallo così costruito contiene il valore vero θ_{vero} se e solo se $v(\theta_{vero}) < \overline{\theta}_{obs} < u(\theta_{vero})$, il che avviene con probabilità α indipendentemente dal particolare valore di θ_{vero}.

Se, come normalmente ci aspettiamo da uno stimatore corretto, le funzioni $v(\theta)$ e $u(\theta)$ sono monotone crescenti e invertibili, la soluzione del problema è

$$\begin{aligned} a &= u^{-1}(\overline{\theta}_{obs}) \\ b &= v^{-1}(\overline{\theta}_{obs}) \end{aligned} \tag{2.48}$$

Se la soluzione non è calcolabile analiticamente, si possono sempre risolvere numericamente per a e b le equazioni

$$\begin{aligned} \beta_1 &= \int_{\overline{\theta}_{obs}}^{\infty} d(\overline{\theta}|\theta = a) d\overline{\theta} \\ \beta_2 &= \int_{-\infty}^{\overline{\theta}_{obs}} d(\overline{\theta}|\theta = b) d\overline{\theta} \end{aligned} \tag{2.49}$$

Dove β_1 e β_2 sono pari, nel caso di intervallo centrale, a $\frac{(1-\alpha)}{2}$.
Nulla vieta di considerare intervalli non centrati, prendendo qualunque valore di β_1 e β_2 tale che $\beta_1 + \beta_2 = 1 - \alpha$. Se ad esempio siamo interessati a determinare un limite superiore per il parametro θ, sceglieremo $\beta_1 = 0$, $\beta_2 = 1 - \alpha$, e la banda di confidenza diventa il semipiano $\overline{\theta} > q_{\overline{\theta}}(1 - \alpha|\theta)$.

2.8.1 Il caso Gaussiano

Nel caso di uno stimatore corretto gaussiano con deviazione standard $\sigma(\overline{\theta})$ nota, la soluzione è particolarmente semplice. Le curve $v(\theta)$ e $u(\theta)$ sono infatti semplicemente rette parallele alla bisettrice:

$$\begin{aligned} v(\theta) &= q_{\overline{\theta}}\left(\frac{1-\alpha}{2}|\theta\right) = \theta - N_\sigma(\alpha)\sigma(\overline{\theta}) \\ u(\theta) &= q_{\overline{\theta}}\left(1-\frac{1-\alpha}{2}|\theta\right) = \theta + N_\sigma(\alpha)\sigma(\overline{\theta}) \end{aligned} \tag{2.50}$$

dove N_σ è ottenuto invertendo l'eq. 1.27

$$N_\sigma = q_{std}\left(\frac{1+\alpha}{2}\right) \tag{2.51}$$

La soluzione 2.48 diventa dunque semplicemente

$$\begin{aligned} a &= \overline{\theta}_{obs} - N_\sigma(\alpha)\sigma(\overline{\theta}) \\ b &= \overline{\theta}_{obs} + N_\sigma(\alpha)\sigma(\overline{\theta}) \end{aligned} \tag{2.52}$$

In questo caso, l'intervallo di confidenza al 68.3% coincide dunque con "l'errore standard" $\overline{\theta}_{obs} \pm \sigma(\overline{\theta})$.
Nel caso in cui $\sigma(\overline{\theta})$ non sia noto a priori, basterà sostituire nell'eq. 2.51 la funzione quantile della gaussiana standard q_{std} con quella della distribuzione di Student (vedi par. 3.6).

2.8.2 Una soluzione approssimata

Come abbiamo visto nel paragrafo 2.6, un qualunque stimatore di massima verosimiglianza diventa gaussiano nel limite di infinite misure $N \to \infty$. Possiamo dunque ottenere il suo intervallo di confidenza dalla formula 2.52, stimando $\sigma(\overline{\theta})$ dal profilo della funzione di likelihood. Si ottengono dunque gli estremi dell'intervallo di confidenza $[a, b]$ dai valori di $\overline{\theta}$ per cui $\mathcal{L}(\overline{\theta})$ ha valore $\mathcal{L}_{max} - N_\sigma(\alpha)^2/2$ (cfr eq. 2.36).
Si può dimostrare che, anche se siamo lontani dal limite asintotico, questo metodo risulta comunque essere una buona approssimazione della soluzione esatta e risulta dunque essere molto utile in pratica, essendo il calcolo molto meno laborioso rispetto al metodo esatto di Neyman. Il metodo permette in particolare di determinare se l'intervallo è asimmetrico rispetto alla miglior stima $\overline{\theta}_{obs}$.

Esempio 2.8.1 *Singola misura da distribuzione esponenziale*
Vogliamo calcolare l'intervallo di confidenza centrale, con livello di confidenza $\alpha = 0.9$, per la stima del parametro τ della distribuzione esponenziale sulla base di una singola misura di t.

Come abbiamo visto, la miglior stima di τ per la distribuzione esponenziale è data dalla media aritmetica, in questo caso l'unico valore osservato t_{obs}. Per ottenere la soluzione esatta dell'intervallo di confidenza col metodo di Neyman, dobbiamo calcolare i quantili $T((1-\alpha)/2)$ e $T((1+\alpha)/2)$ come funzione di τ risolvendo l'equazione

$$P(T(q)) = q$$

dove

$$P(t) = 1 - e^{-t/\tau}$$

è la distribuzione di probabilità cumulativa. Si trova dunque

$$T(q;\tau) = -\tau \log(1-q)$$

Gli estremi dell'intervallo sono infine ottenuti da

$$T((1+\alpha)/2;\tau_{min}) = t_{obs}$$

$$T((1-\alpha)/2;\tau_{max}) = t_{obs}$$

da cui

$$-\frac{t_{obs}}{\log((1-\alpha)/2)} < \tau < -\frac{t_{obs}}{\log((1+\alpha)/2)}$$

Si noti che l'intervallo è asimmetrico, poichè la miglior stima resta $\tau = t_{obs}$.

Possiamo confrontare questa soluzione con la stima "standard"

$$\tau = t_{obs} \pm N_\sigma(\alpha) \cdot t_{obs}$$

(si ricordi che $\sigma^2(t) = \tau^2$), dove N_σ è il numero di deviazioni standard che corrisponde all'intervallo di confidenza α nel caso gaussiano, ed è ricavabile con R come

```
> alpha=0.9
> qnorm((1+alpha)/2)
[1] 1.644854
```

Infine, possiamo utilizzare il metodo approssimato tramite la funzione di massima verosimiglianza, risolvendo l'equazione

$$-\log L(\tau_{min,max}) = -\log L(t_{obs}) + \frac{N_\sigma^2(\alpha)}{2}$$

ovvero

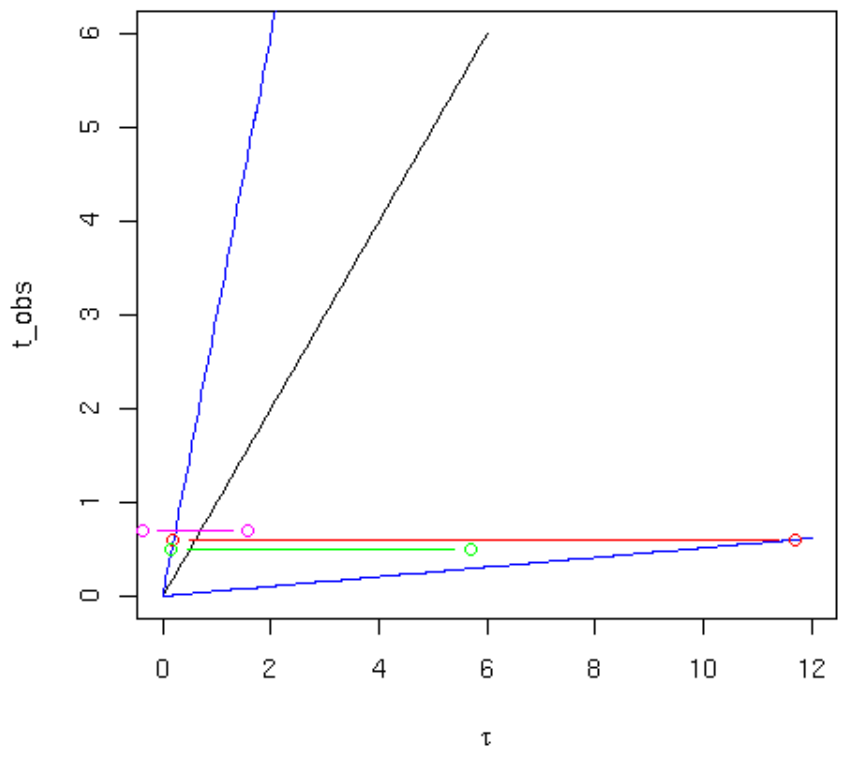
$$\log \tau_{min,max} + \frac{t_{obs}}{\tau_{min,max}} = \log t_{obs} + 1 + \frac{N_\sigma^2(\alpha)}{2}$$

che possiamo risolvere numericamente tramite il seguente codice, che mostra anche un confronto grafico fra i tre intervalli ottenuti:

```
mloglike =function(tau=1,tobs) {
 log(tau)+tobs/tau
}

nsig = function(alpha) {
 qnorm((alpha+1)/2)
}

confneyman = function(tobs,alpha) {
  c( -tobs/log((1-alpha)/2) ,
```

```
     tobs,
     -tobs/log((1+alpha)/2))
}

confstd = function(tobs,alpha) {
 terr = nsig(alpha) * tobs
 c( tobs - terr,
    tobs,
    tobs + terr)
}

confll = function(tobs,alpha) {
# soluzione ML: partendo dal massimo, ci spostiamo di una quantita'
# piccola (rispetto all'errore) step, fino a trovare il valore per cui
# loglik varia di nsig^2/2
 deltal=(nsig(alpha))^2/2
 llmin=mloglike(tobs,tobs)
 step = (nsig(alpha)*tobs)/20

 tminll=tobs
 while (mloglike(tminll,tobs) < llmin+deltal) {
  tminll = tminll - step
  # protezione contro log(numero negativo)
  if(tminll<0) break
 }
 tmaxll=tobs
 while(mloglike(tmaxll,tobs) < llmin+deltal) {
   tmaxll = tmaxll + step
 }
 c( tminll,
    tobs,
    tmaxll)
}


seeconf = function(alpha=0.9,tobs=0.6) {
# disegna il diagramma di Neyman
 curve(x*1,0,10*tobs,xlab=expression(tau),ylab=expression(t_obs),xlim=c(0,20*tobs))
 curve(-x*log((1-alpha)/2),add=T,col="blue",0,10*tobs)
 curve(-x*log((1+alpha)/2),add=T,col="blue",0,20*tobs)

# soluzione esatta
 ney = confneyman(tobs,alpha)
 cat("tau is ",ney[2]," - ",(ney[2]-ney[1])," + ",(ney[3]-ney[2]),"\n")
 lines(c(ney[1],ney[3]),c(ney[2],ney[2]),col="red",type="b")

# soluzione standard
 std = confstd(tobs,alpha)
```

```
 cat("tau standard is ",std[2]," +/- ",std[2]-std[1],"\n")
 lines(c(std[1],std[3]),c(std[2]+0.1,std[2]+0.1),col="magenta",type="b")

# soluzione ML
 ll = confll(tobs,alpha)
 cat("taull is ",ll[2]," - ",ll[2]-ll[1]," + ",ll[3]-ll[2],"\n")
 lines(c(ll[1],ll[3]),c(ll[2]-0.1,ll[2]-0.1),col="green",type="b")
}

> seeconf()
tau is           0.6  - 0.3997151  + 11.09744
tau standard is  0.6  +/-  0.9869122
taull is         0.6  - 0.464758  + 5.112338
```



Per quantificare la differenza fra il significato dei tre intervalli, possiamo calcolarne il "ricoprimento", ovvero l'effettivo livello di confidenza. E' necessario per questo simulare un gran numero di esperimenti e contare quante volte l'intervallo ottenuto contiene il valore vero di τ usato nella simulazione. Per l'intervallo ottenuto col metodo esatto ci aspettiamo naturalmente di trovare un ricoprimento pari ad α.

```
coverage = function(n=1000,tau=1,alpha=0.9) {
# simuliamo gli n esperimenti, si noti che il valore di tau
# e' ininfluente sul risultato
 xx=rexp(n,rate=1/tau)
```

```
 nokney=0
 nokst=0
 nokll=0
 for (i in 1:n) {
  tobs=xx[i]

# intervallo di neyman
  ney = confneyman(tobs,alpha)
  if(tau >= ney[1] & tau <= ney[3]) {
   nokney=nokney+1
  }

# intervallo standard
  std = confstd(tobs,alpha)
  if(tau >= std[1] & tau <= std[3]) {
   nokst=nokst+1
  }

# intervallo ML
  ll = confll(tobs,alpha)
  if(tau >= ll[1] & tau <= ll[3]) {
   nokll=nokll+1
  }
 }

# calcoliamo il ricoprimento e il suo errore (binomiale)
 covney=nokney/n
 dcovney=sqrt(covney*(1-covney)/n)
 cat("coverage Neyman =",covney," +/- ",dcovney,"\n")

 covst=nokst/n
 dcovst=sqrt(covst*(1-covst)/n)
 cat("coverage standard=",covst," +/- ",dcovst,"\n")

 covll=nokll/n
 dcovll=sqrt(covll*(1-covll)/n)
 cat("coverage ML=",covll," +/- ",dcovll,"\n")

}

> coverage(n=20000,alpha=0.9)
coverage Neyman   = 0.900   +/-  0.002
coverage standard = 0.681   +/-  0.003
coverage ML       = 0.875   +/-  0.002
```

Capitolo 3

Tests statistici di ipotesi

In questo capitolo verranno trattati vari tests statistici, ovvero analisi statistiche mirate a verificare la compatibilità di una data ipotesi con un campione di dati.

3.1 Test di un'ipotesi: significatività e p–value

Indicheremo con H_0 l'ipotesi teorica che vogliamo testare, detta **ipotesi nulla**. Un test consiste nel ricavare dal campione $\mathcal{S}$ una **statistica di test** $T(\mathcal{S})$, ovvero una o più variabili che caratterizzano nel modo più appropriato la compatibilità dei dati con l'ipotesi nulla. Si vuole cioè che i valori assunti da T nell'ipotesi H_0, descritti dalla PDF $\mathcal{d}_0(T|H_0)$, siano il più possibile separati dai valori che T assumerebbe se H_0 non fosse vera. Si separa poi lo spazio dei possibili valori di T in una **regione accettata** A ed una **regione critica** C, tali che

$$\begin{aligned} \int_A \mathcal{d}_0(T|H_0)dT &= 1 - \alpha_s \\ \int_C \mathcal{d}_0(T|H_0)dT &= \alpha_s \end{aligned} \tag{3.1}$$

Si dice che il test ha esito negativo (i dati sono compatibili con l'ipotesi nulla H_0) se $T \in A$ e esito positivo (i dati non sono compatibili con H_0) se $T \in C$. La terminologia è dovuta al fatto che i tests sono spesso usati per vedere se dai dati è possibile mettere in evidenza un effetto cercato (ad esempio, l'effetto benefico di un farmaco o di un regime alimentare, oppure la presenza di una malattia dai risultati di un test clinico): l'ipotesi più semplice da formulare è quella che non ci sia nessun effetto (ipotesi "nulla") e dunque conviene testare tale ipotesi e si dirà che l'esito è positivo se l'effetto è stato messo in evidenza.
Il valore α_s, detto **significatività** del test, rappresenta la probabilità di avere un "errore di primo tipo" (o "falso positivo"), ovvero di ottenere un risultato positivo anche se l'ipotesi nulla è vera. Si sceglie dunque la regione accettata in modo da avere un piccolo ($\ll 0.5$, se vogliamo che il test abbia un senso) valore di α_s. Quanto piccolo, dipende dal caso in esame, come vedremo negli esempi di questo capitolo.
Si noti che l'esito negativo di un test non implica necessariamente che l'ipotesi nulla sia vera, ma solo che non possiamo escluderla sulla base dei nostri dati. In caso di esito positivo del test,

si dice che l'ipotesi nulla è esclusa con significatività α_s, oppure con confidenza $1-\alpha_s$, tenendo sempre presente che, se l'ipotesi nulla fosse vera, avremmo una probabilità α_s di sbagliarci.

Se vogliamo, anziché avere semplicemente un risultato binario (positivo/negativo), quantificare meglio l'aderenza del campione di dati all'ipotesi nulla, è utile introdurre il *p–value*, definito come la probabilità, nell'ipotesi nulla, di ottenere un valore di T uguale o "peggiore" di quello ottenuto. La definizione di "peggiore" dipende dall'ipotesi altenativa che ci aspettiamo. Se ad esempio la statistica di test è un numero reale, e ci aspettiamo che l'ipotesi H_0 risulti in valori di T più bassi rispetto a qualunque ipotesi alternativa, faremo un "test a una coda", ovvero tratteremo come "peggiori" tutti i valori maggiori di quello osservato:

$$p\text{–}value_{1+} = \int_{T_{obs}}^{\infty} d_0(T|H_0)dT \tag{3.2}$$

Se invece ci aspettiamo valori più alti, faremo l'integrale sull'altra coda della distribuzione:

$$p\text{–}value_{1-} = \int_{-\infty}^{T_{obs}} d_0(T|H_0)dT \tag{3.3}$$

Infine, se le ipotesi alternative possono dare qualunque valore di T, faremo un "test a due code":

$$p\text{–}value_2 = \int_{|T-E(T|H_0)|\geq|T_{obs}-E(T|H_0)|} d_0(T|H_0)dT \tag{3.4}$$

Chiaramente il test ha esito positivo se $p\text{–}value < \alpha_s$.

Se vogliamo testare H_0 contro un'ipotesi alternativa H_1, possiamo definire

$$\beta_1 = \int_A d_1(T|H_1)dT \tag{3.5}$$

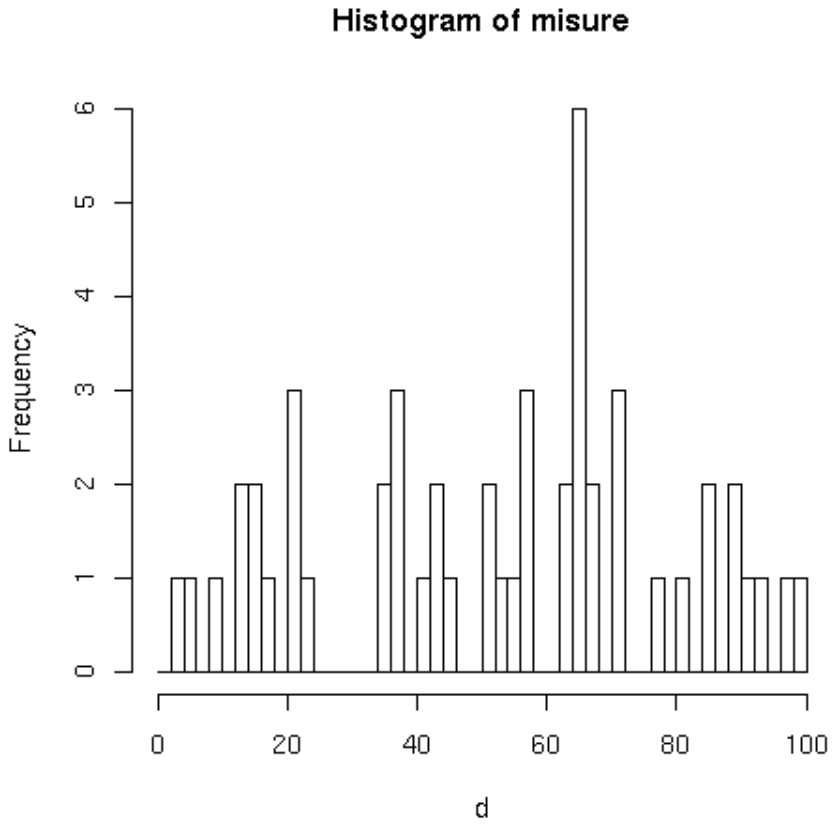
come la probabilità di avere un "errore di secondo tipo" (o "falso negativo"), ovvero un risultato negativo del test nell'ipotesi che H_1 sia vera (e dunque H_0 falsa). Possiamo diminuire α_s aumentando la regione accettata A, ma questo comporta un aumento di β_1. La scelta della regione accettata risulta dunque da un compromesso fra i due tipi di errore e dipende dal caso in esame.
La quantità $1-\beta_1$ è detta **potenza** del test, essendo il criterio, a parità di α_s, con cui si confrontano le prestazioni di tests diversi.

È importante sottolineare che la scelta della regione accettata deve essere fatta **prima** di osservare i dati, ovvero non deve essere influenzata dai dati stessi, pena la perdita del significato statistico del *p–value*, che risulterebbe distorto. Se non è possibile calcolare a priori $d_0(T|H_0)$ (sulla base, ad esempio, di una legge della fisica), si sceglierà la regione accettata sulla base di un campione di dati simulato, oppure sulla base di un campione di dati (detto "di training") indipendente da quello su cui si effettua l'analisi.

Esempio 3.1.1 *Una falsa scoperta*
I valori nel file

/afs/math.unifi.it/service/Rdsets/ricercasegnale.rdata
rappresentano i risultati di un esperimento di ricerca di vita extraterrestre. I valori rappresentano, in unità arbitrarie, la coordinata spaziale di provenienza dei candidati osservati. Sappiamo che ci si aspetta un fondo di falsi segnali, indipendenti fra loro e uniformemente distribuiti nella nostra unità, pari 1.04 oggetti/intervallo. Vogliamo allora definire un test statistico in cui l'ipotesi nulla è l'assenza di un segnale reale con $\alpha_s = 1\%$. In caso di esito positivo del test, potremo annunciare di aver scoperto un segnale di vita extraterrestre con una confidenza del 99%.
Osserviamo i dati facendo un istogramma:

```
misure = scan("/afs/math.unifi.it/service/Rdsets/ricercasegnale.rdata")
h=hist(misure,breaks=seq(0,100,2))
```



Notiamo un picco sospetto: 6 conteggi nell'intervallo 64–66. Potremmo allora pensare di formulare il test come segue: utilizziamo il valore di conteggi in questo intervallo come statistica di test, sapendo che nell'ipotesi nulla questa segue la distribuzione di Poisson con valore atteso 1.04. Sapendo che l'osservazione di un segnale reale implica un valore più alto di conteggi, faremo il test a una coda prendendo come *p–value*

$$\delta = \sum_{k \geq k_{obs}} f_P(k; \lambda) = 1 - p_P(k_{obs} - 1; \lambda)$$

Il valore numerico è

```
> 1-ppois(5,1.04)
[1] 0.0007269856
```

e concluderemmo quindi di aver scoperto E.T. con una confidenza del 99.93%.
Tuttavia l'impostazione di questo test è errata, in quanto esso è stato definito sulla base dei dati osservati, avendo scelto i conteggi da un solo intervallo, che è quello col valore osservato più alto. Per impostare il test correttamente, la PDF della statistica di test deve rappresentare la probabilità di ottenere k_{obs} conteggi come valore massimo sugli n=50 conteggi osservati, ciascuno dei quali segue la statistica di Poisson $f_P(k_i; 1.04)$. Il *p–value* si ottiene dunque come il complementare della probabilità che tutti gli intervalli abbiano meno di k_{obs} conteggi, ovvero

$$p - value = 1 - p_P(k_{obs} - 1; \lambda)^n = 1 - (1 - \delta)^n = 0.036$$

un valore compatibile, entro la nostra significatività, con l'ipotesi nulla.

3.2 Il test χ^2

Il **test di Pearson**[11], o test χ^2, è probabilmente il test statistico più comunemente utilizzato. Nei capitoli precedenti abbiamo spesso confrontato la distribuzione dei valori di una variabile in un campione di dati con i valori attesi in base alla distribuzione ipotizzata. Abbiamo fatto questo confronto in modo qualitativo, sovrapponendo il grafico della funzione densità aspettata, moltiplicata per l'opportuno fattore di normalizzazione, con l'istogramma dei valori sperimentali, in modo da confrontare visivamente la forma delle distribuzioni. Il test χ^2 permette di fare questo confronto, detto "test di bontà del fit", in modo quantitativo.

Supponiamo di avere un istogramma con n intervalli della variabile osservata x e indichiamo con k_i il conteggio in ciascun intervallo I_i. L'ipotesi nulla del test è che i dati seguano la distribuzione $d(x)$, per cui il valore atteso di ciascun k_i è dato da

$$\lambda_i = N \cdot \int_{x \in I_i} d(x) dx \simeq N \delta_i d(\overline{x_i}) \tag{3.6}$$

dove N è il numero di valori attesi nel campione, $\overline{x_i}$ e δ_i sono il valore centrale e la larghezza dell'intervallo e l'ultimo passaggio è giustificato solo se $d(x)$ può essere approssimata a una funzione lineare all'interno dell'intervallo.

Nell'usuale ipotesi di dati indipendenti, ciascuna variabile aleatoria k_i segue una distribuzione di Poisson (o binomiale, se N è fissato). Se i valori di k_i sono sufficientemente grandi (in pratica $\gtrsim 5$) possiamo approssimare la distribuzione a una gaussiana che, nell'ipotesi nulla, ha valore atteso e varianza pari a λ_i. In questa ipotesi si avrà allora che

$$\chi^2 = \sum_{i=1}^{n} \frac{(k_i - \lambda_i)^2}{\lambda_i} \tag{3.7}$$

segue la distribuzione di Pearson (cfr par. 1.14). Il numero di gradi di libertà è pari a $n_{df} = n$ se i valori di λ_i non dipendono dai valori osservati. Nel frequente caso in cui la nostra ipotesi

riguardi solo la forma della distribuzione $d(x)$, mentre il valore N è ricavato dal numero di valori osservato, si dimostra (vedi [1]) che il χ^2 segue ancora la distribuzione di Pearson, ma con $n_{df} = n - 1$ gradi di libertà. Se poi $d(x)$ dipende da m_p parametri che stimiamo dai dati stessi, si avrà $n_{df} = n - m_p - 1$, il che spiega l'espressione "gradi di libertà".

Poiché qualunque ipotesi alternativa implica valori di χ^2 maggiori rispetto all'ipotesi nulla, il test viene fatto ad una coda:

$$p - value = \int_{\chi^2}^{\infty} f_{\chi^2}(x; n_{df})dx = 1 - p_{\chi^2}(\chi^2; n_{df}) \tag{3.8}$$

Si noti che, nelle approssimazioni fatte, la distribuzione del χ^2 è indipendente dalla $d(x)$, e questo rende il test di facile implementazione, il che spiega la sua popolarità.

È utile sottolineare che, essendo le fluttuazioni dei conteggi ineliminabili, un valore del χ^2 molto inferiore a quello atteso (n_{df}), ovvero un *p–value* molto vicino a 1, pur risultando in un esito negativo del test, è il sintomo di un errore nell'analisi. Ad esempio, i dati potrebbero non essere indipendenti. Bisogna poi fare attenzione a non confondere il numero di intervalli n, da cui si ricava il numero di gradi di libertà, con il numero N dei valori nell'istogramma!

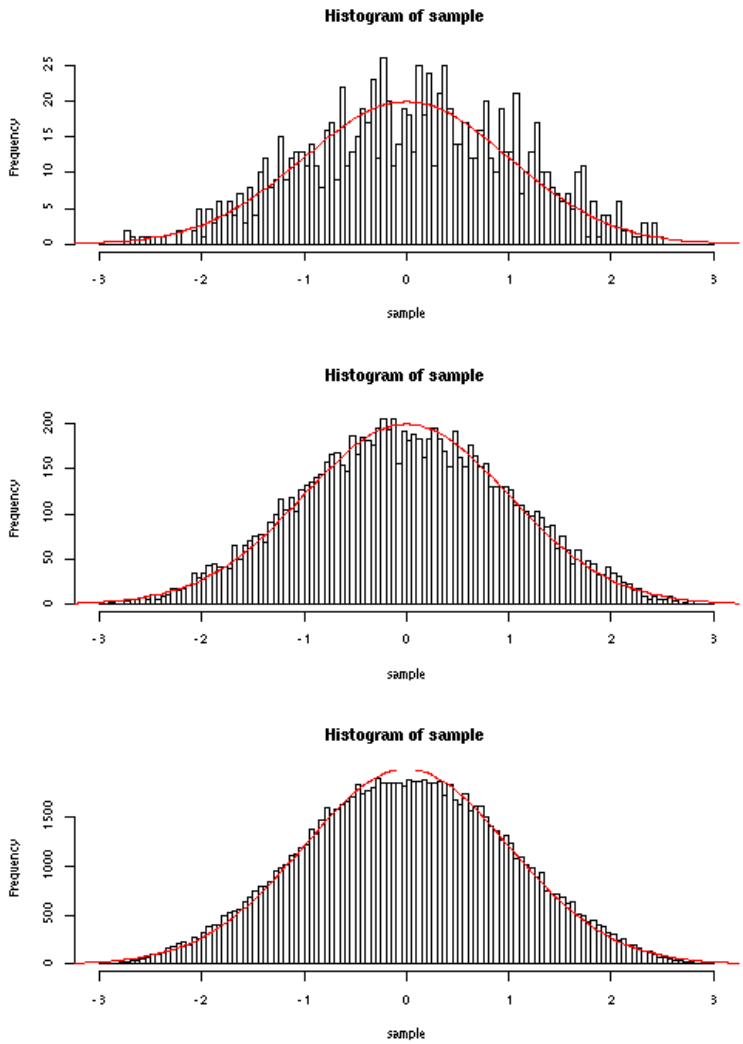
Esempio 3.2.1 *Test di normalità tramite* χ^2
Abbiamo visto come, in virtù del teorema del limite centrale, la distribuzione densità della somma di n variabili distribuite uniformemente tenda rapidamente, al crescere di n, ad assomigliare a una distribuzione normale (esempio 1.13.1). Vogliamo ora rendere più quantitativo questo confronto, eseguendo un test χ^2 per verificare l'ipotesi gaussiana su un campione di N valori simulati:

```
mychisqunif = function(n=3,N=1000) {
 sample = runif(N,-1,1)
 if (n >1) {
  for (i in 1:(n-1)) {
   sample = sample + runif(N,-1,1)
  }
 }
 sigmacentral=2*sqrt(n/12)
 delta=sigmacentral/20
 h=hist(sample,breaks=seq(-1*n,1*n,delta))
 curve(dnorm(x,sd=sigmacentral)*N*delta,add=T,col="red")

 x=h$mids
 dati=h$counts
 attesi = dnorm(x,sd=sigmacentral)*N*delta
 chi2 = sum((dati-attesi)^2/attesi)
 ngl = length(x) - 1
 cat("chi2= ",chi2," dof=",ngl,"   p-value is ",1-pchisq(chi2,df=ngl),"\n")
}
```

Il numero di gradi di libertà è in questo caso pari al numero di intervalli meno uno, poichè nel calcolo dei valori attesi N è fissato alla dimensione del campione, mentre i due parametri della

gaussiana erano fissati a priori dalla predizione del teorema del limite centrale. Se avessimo voluto fare un test di normalità senza alcuna ipotesi sul valore atteso e la deviazione standard, avremmo potuto stimare questi parametri dai dati stessi e ridurre di due il numero di gradi di libertà.

Con i valori $n = 3$, $N = 1000$, il test dà tipicamente valori del *p–value* alti che non ci permettono di escludere l'ipotesi che il campione segua la distribuzione normale. Per poter "accorgersi" dai dati che la reale distribuzione non è una gaussiana, è necessario disporre di statistiche molto superiori:




```
> par(mfrow=c(3,1))
> mychisqunif(3,1000)
chi2=  138.0771  dof= 119    p-value is  0.1114694
> mychisqunif(3,10000)
chi2=  157.4705  dof= 119    p-value is  0.01048201
> mychisqunif(3,100000)
chi2=  703.2427  dof= 119    p-value is  0
```

Aumentando il valore di n, la statistica necessaria per far fallire il test aumenta ulteriormente.

Esempio 3.2.2 *Test di ipotesi poissoniana per conteggi di raggi cosmici*
Nell'esempio 2.2.3 abbiamo considerato i dati di un reale esperimento di conteggio di raggi cosmici, e verificato visivamente che il numero di eventi osservati in un intervallo di 2 minuti segue una ditribuzione di Poisson, come ci si attende nell'ipotesi di independenza degli eventi. Facciamo ora un confronto quantitativo per mezzo del test χ^2:

```
chisqcosmic = function() {
 df = read.table(file="CosmoData.txt",skip=7)
 c = df$V3
 cm=mean(c)
 n=length(c)
# costruiamo l'istogramma con bins di larghezza 1 centrati nei valori interi
 xmin=min(c)-1.5
 xmax=max(c)+1.5
 x=xmin:xmax
 h=hist(c,breaks=x)
 xp=h$mids
 dati=h$counts
 attesi = dpois(xp,lambda=cm)*n
 points(xp,attesi,type="p",col="red",pch=19)
 chi2 = sum((dati-attesi)^2/attesi)
 nbin=length(x)
 ngl = nbin - 2
 cat("chi2= ",chi2," dof=",ngl,"   p-value is ",1-pchisq(chi2,df=ngl),"\n")
}
```

Si noti che in questo caso il numero di gradi di libertà è n-2, poichè abbiamo usato i dati, oltre che per normalizzare la predizione, anche per stimarne il valore atteso.

Dal risultato *p–value*=65% concludiamo che i dati sono del tutto compatibili con l'ipotesi poissoniana.

Nel caso in cui il valore dei conteggi sia troppo basso perché l'approssimazione gaussiana sia valida, il χ^2 non seguirà più la distribuzione di Pearson, ma una distribuzione che dipende dalla PDF $\mathcal{d}(x)$. Il test può essere ancora eseguito, ma perde la sua praticità, in quanto il *p–value* dovrà essere calcolato ricavando, tipicamente tramite una simulazione, la distribuzione attesa di χ^2 per lo specifico caso in esame.

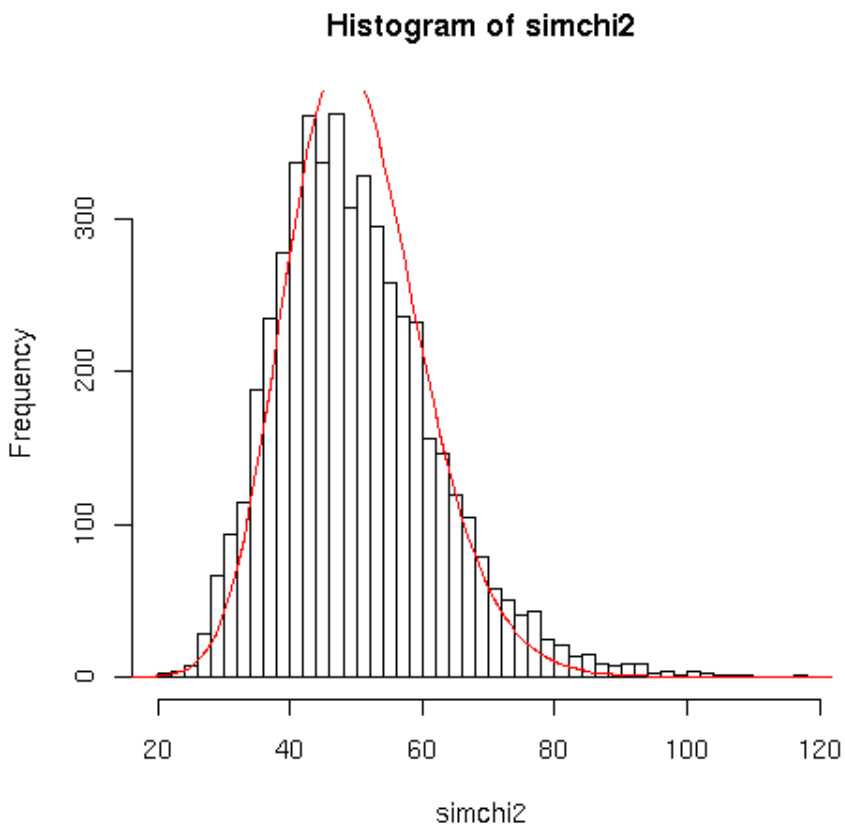
Esempio 3.2.3 *Significatività di un debole segnale in presenza di fondo*
Ritorniamo al nostro esperimento di ricerca di vita extraterrestre dell'esempio 3.1.1. Testiamo adesso l'ipotesi nulla (assenza di segnale oltre il fondo) utilizzando il test χ^2:

```
> misure = scan("/afs/math.unifi.it/service/Rdsets/ricercasegnale.rdata")
> h=hist(misure,breaks=seq(0,100,2))
> dati=h$counts
```

```
> ngl = length(dati)
> fondo.atteso=1.04 # per bin
> attesi=rep(fondo.atteso,ngl)
> chi2 = sum((dati-attesi)^2/attesi)
> cat("chi2= ",chi2," dof=",ngl,"   p-value is ",1-pchisq(chi2,df=ngl),"\n")
chi2=  68.26923  dof= 50    p-value is  0.0438842
```

Si noti che in questo caso il numero di gradi di libertà coincide col numero di intervalli, essendo il valore atteso specificato a priori. Otteniamo un *p–value* pari al 4.3%. Tuttavia, dato il basso valore dei conteggi, non possiamo aspettarci che la distribuzione di Pearson sia una buona approssimazione per la distribuzione del nostro χ^2. Calcoliamo il valore corretto del *p–value* simulando un gran numero di analoghi esperimenti nell'ipotesi nulla.

```
lowsignal = function(Nsim=5000) {
 misure = scan("/afs/math.unifi.it/service/Rdsets/ricercasegnale.rdata")
 h=hist(misure,breaks=seq(0,100,2))
 dati=h$counts
 ngl = length(dati)
 fondo.atteso=1.04 # per bin
 attesi=rep(fondo.atteso,ngl)

# simuliamo Nsim esperimenti con solo fondo e mettiamo nel vettore
# simchi2 i valori del chi2 ottenuto in ciascun esperimento
 simchi2=c()
 for (i in 1:Nsim) {
   simset=rpois(ngl,lambda=fondo.atteso)
   simchi2[i]=sum((simset-attesi)^2/attesi)
 }

# calcoliamo la frazione di esperimenti simulati con un valore di chi2
# superiore a quello osservato
 np=length(simchi2[simchi2>=chi2])
 pv=np/Nsim
 dpv=sqrt(pv*(1-pv)/Nsim)
 cat("correct p-value is ",pv," +/- ",dpv,"\n")
# confrontiamo la distribuzione di chi2 simulata
# con quella di Pearson
 ch=hist(simchi2,breaks=50)
 delta=ch$breaks[2]-ch$breaks[1]
# confrontiamo la distribuzione ottenuta co quella di Pearson
# usata in precedenza
 curve(dchisq(x,df=ngl)*Nsim*delta,add=T,col="red")
}

> lowsignal()
correct p-value is  0.085  +/-  0.004
```



Notiamo che la reale distribuzione del χ^2 ha code più importanti rispetto a quella attesa nel limite di grandi conteggi, per cui il valore corretto del *p–value* è sensibilmente maggiore di quanto stimato in precedenza.

3.3 Il test di Kolmogorov

Un test di bontà del fit più adatto al caso di bassi conteggi è quello di Kolmogorov–Smirnov. Consiste nel confrontare la distribuzione di probabilità cumulativa attesa $p(x)$ con quella empirica

$$S(x) = \frac{\sum_{i=1}^{n} \chi(x_i < x)}{N} \tag{3.9}$$

dove $\chi(x_i < x)$ è la funzione indicatrice che vale 1 se $(x_i < x)$ e 0 altrimenti, e N è il numero totale di valori del campione. Si calcola la differenza massima

$$D = max \left| S(x) - p(x) \right| \tag{3.10}$$

Nell'ipotesi nulla la quantità $\kappa = \sqrt{N} D$ diventa indipendente dalla distribuzione $p(x)$. La sua funzione di ripartizione, nel limite $n \to \infty$ tende alla funzione di Kolmogorov

$$p_{KS}(\kappa) = 1 - 2 \sum_{j=1}^{\infty} (-1)^{j-1} e^{-2j^2\kappa^2} = \frac{\sqrt{2\pi}}{\kappa} \sum_{j=1}^{\infty} e^{-(2j-1)^2\pi^2/(8\kappa^2)} \tag{3.11}$$

Come nel caso del test χ^2, si esegue il test a una coda:

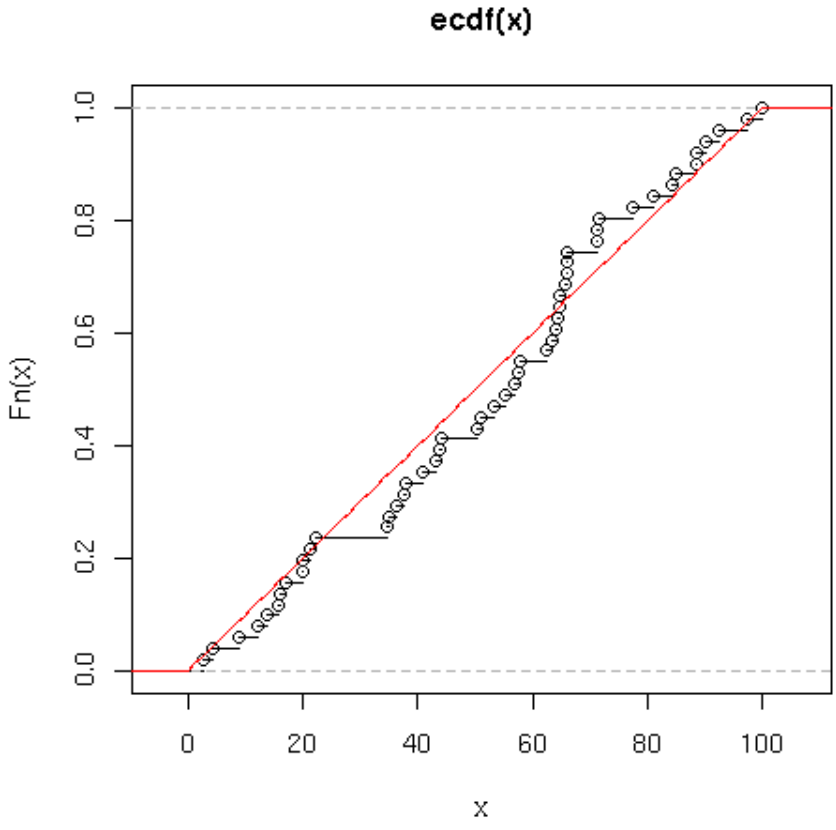
$$p\text{–}value = 1 - p_{KS}(\kappa) \tag{3.12}$$

Il software R permette, tramite la funzione ks.test(), di calcolare il p–value esatto anche nel caso di bassi valori di n. La funzione cumulativa empirica può essere ottenuta dalla funzione plot.ecdf().

Il test di Kolmogorov è utile per testare la forma della distribuzione, ma non i valori assoluti dei conteggi. Inoltre, il test non è più valido se la predizione $p(x)$ dipende dai dati stessi, ad esempio per un test di normalità in cui si stimino dai dati il valore atteso e la varianza della gaussiana. Esistono molte varianti per trattare questi casi (test di Lilliefors [4], test di Kuiper, test di Anderson-Darling...).

Esempio 3.3.1 *Ancora sulla significatività di un debole segnale*
Ripetiamo nuovamente l'analisi dell'esempio 3.2.3 utilizzando stavolta il test di Kolmogorov–Smirnov per verificare la compatibilità con una distribuzione uniforme.
Confrontiamo innanzi tutto la funzione di probabilità cumulativa empirica del campione $S(x)$ con la probabilità cumulativa attesa per la distribuzione uniforme:

```
misure = scan("/afs/math.unifi.it/service/Rdsets/ricercasegnale.rdata")
plot.ecdf(misure)
curve(punif(x,min=0,max=100),add=T,col="red")
```



Il parametro D del test, che è la massima distanza fra le due curve nel grafico, e il corrispondente *p–value* sono ottenibili dalla funzione *ks.test()*:

```
> ks.test(misure,punif,min=0,max=100)

        One-sample Kolmogorov-Smirnov test

data:  misure
D = 0.112, p-value = 0.5084
alternative hypothesis: two.sided
```

Secondo questo test, i dati sono dunque ampiamente compatibili con l'ipotesi nulla.

3.4 z Test

Supponiamo di avere uno stimatore $\overline{\theta}$ che sappiamo seguire una distribuzione normale con deviazione standard $\sigma_{\overline{\theta}}$ (ad esempio, uno stimatore ML con N $\gg 1$). Vogliamo testare l'ipotesi che il valore atteso di $\overline{\theta}$ sia pari a θ_0. Nell'ipotesi nulla, la quantità

$$z = \frac{(\overline{\theta} - \theta_0)}{\sigma_{\overline{\theta}}} \tag{3.13}$$

segue la distribuzione normale standard. Possiamo allora effettuare un test a due code, detto "z–test", definendo la regione accettata come

$$\begin{aligned} q_{std}(\alpha_s/2) < z < q_{std}(1-\alpha_s/2) \\ \Longrightarrow \quad |z| < q_{std}(1-\alpha_s/2) \end{aligned} \tag{3.14}$$

Il *p–value* sarà

$$p\text{–}value = 2(1 - p_{std}(|\overline{\theta}|)) \tag{3.15}$$

dove p_{std} e q_{std} sono la funzione di ripartizione e la funzione quantile della gaussiana standard.

Se vogliamo efefttuare il test con ipotesi alternativa è $\theta > \theta_0$, o $\theta < \theta_0$, ci conviene effettuare un test a una coda. Le formule per la regione accettata, il *p–value*, e il corrispondente intervallo di confidenza per θ (cfr. par. 2.8) sono riassunte in tabella 3.1. Si noti che il test ha esito positivo con significatività α_s quando l'intervallo di confidenza con livello $\alpha_{CL} = 1 - \alpha_s$ non contiene il valore θ_0.

Ipotesi alt.	Regione accetata	*p–value*	Intervallo di confidenza
2 code	$\lvert z\rvert < q_{std}(1-\alpha_s/2)$	$2(1 - p_{std}(\lvert\overline{\theta}\rvert))$	$\theta = \overline{\theta} \pm \sigma_{\overline{\theta}} q_{std}((1+\alpha_{CL})/2)$
$\theta > \theta_0$	$z < q_{std}(1-\alpha_s)$	$1 - p_{std}(z)$	$\theta > \overline{\theta} - \sigma_{\overline{\theta}} q_{std}(\alpha_{CL})$
$\theta < \theta_0$	$z > q_{std}(\alpha_s)$	$p_{std}(z)$	$\theta < \overline{\theta} + \sigma_{\overline{\theta}} q_{std}(1-\alpha_{CL})$

Tabella 3.1: Formule per lo z–test

3.5 Runs Test

Il *Runs test* è utilizzato per valutare in modo quantitativo se una sequenza di valori di una variabile binaria X (con valori 0 o 1) è compatibile con l'essere una sequenza casuale, ovvero una serie di valori estratti indipendentemente con la stessa probabilità $P(X = 1)$.

Indichiamo con N_1 e N_0 il numero di valori con $X = 1$ o 0, e con R il numero di *runs*, ovvero di sequenze in cui il risultato si ripete. Ad esempio, la sequenza

001110110100

ha 7 *runs*.
Si dimostra che

$$E(R) = \frac{N + 2N_0N_1}{N} \tag{3.16}$$

$$\sigma^2(R) = \frac{2N_0N_1(2N_0N_1 - N)}{N^2(N-1)} = \frac{(E(R)-1)(E(R)-2)}{N-1} \tag{3.17}$$

e che nel limite $N \to \infty$ la variabile R tende ad essere normale. Possiamo dunque eseguire uno z–test a due code con

$$z_{runs} = \frac{(R - E(R))}{\sigma(R)} \tag{3.18}$$

tenendo conto del fatto che il *p–value* ottenuto è esatto solo nel limite di infinita statistica.
Il *runs test* può essere usato in più modi:

testare l'indipendenza dei dati in un campione: l'ipotesi 2.2 che i dati (x_i) in un campione siano indipendenti ed identicamente distribuiti può essere testata ponendo

$$X = \begin{cases} 1 & x_i \geq \overline{x} \\ 0 & x_i < \overline{x} \end{cases} \tag{3.19}$$

test di bontà del fit: ponendo

$$X = \begin{cases} 1 & k_i \geq \lambda_i \\ 0 & k_i < \lambda_i \end{cases} \tag{3.20}$$

si ottiene un test di bontà del fit sensibile ai segni e non ai valori assoluti degli scarti fra i conteggi k_i e i valori attesi λ_i, e dunque alternativo , seppur meno potente, al test χ^2 (che è sensibile agli scarti quadratici).

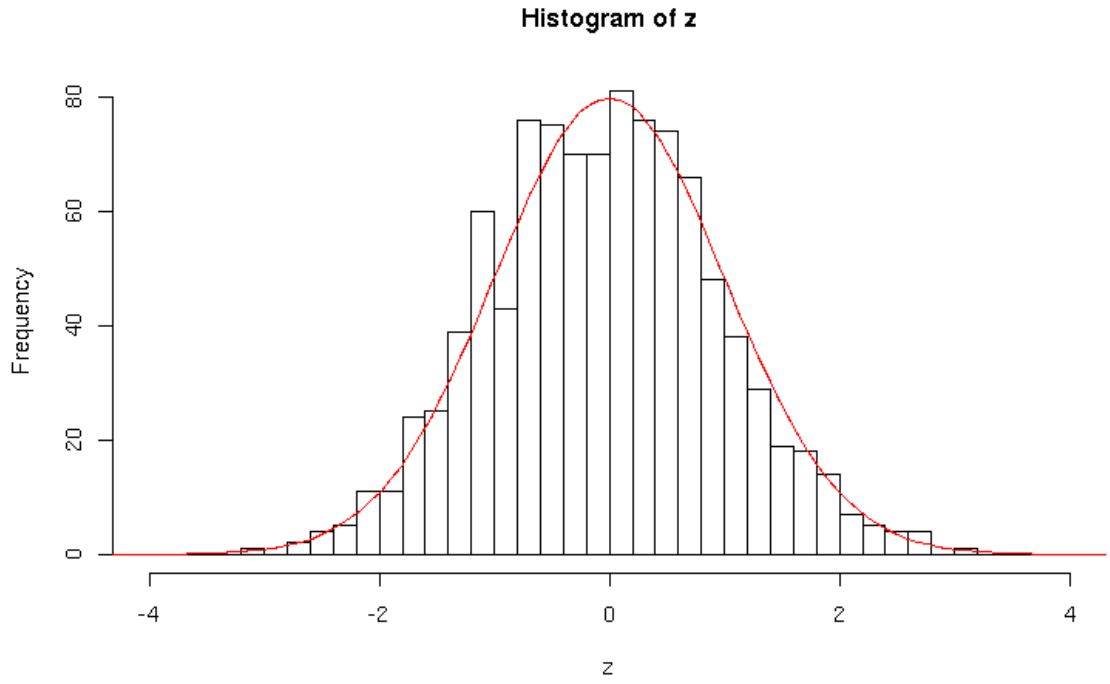
Esempio 3.5.1 *Test di un generatore MonteCarlo*

Verifichiamo, tramite un *Runs test* con significatività del 5%, che le sequenze di valori ottenute dal generatore pseudo–random $runif()$ sono compatibili con una sequenza casuale. Implementiamo il test in R nella funzione *runstest()* ed eseguiamolo su una sequenza ottenuta da $runif()$.

```
runstest = function(data,speak=T) {
 R=0
 b=-1
 na=0
 nb=0
 for (i in 1:length(data)) {
  if (data[i]) {
   ib=1
   nb=nb+1
  }
  else {
   ib=0
   na=na+1
  }
  if(ib != b) { R=R+1 }
  b=ib
 }
 n=na+nb
 expR=(n+2*na*nb)/n
 sigmaR=sqrt( (expR-1)*(expR-2)/(n-1) )
 z= (R-expR)/sigmaR
 if(speak) {
  cat("observed R=",R,", expected R=",expR,", expected sigmaR=",sigmaR,"\n")
  cat("z=",z,"  p-value=",2*(1-pnorm(abs(z)) ),"\n") }
 z
}
> set.seed(99)
> runstest(runif(1000) >= 0.5)
z= -0.4328301   p-value= 0.6651382
```

Il test ha dunque, per questa sequenza, un esito negativo.
Possiamo anche verificare, ripetendo il test molte volte, che, con la statitica del campione scelta pari a 1000, z_{runs} segue la distribuzione gaussiana standard:

```
runifrunstest = function(Nsim=1000,Nsample=1000) {
 z=c()
 for (i in 1:Nsim) {
  z[i]=runstest(runif(Nsample)>0.5 , speak=F)
 }
 hist(z,breaks=seq(-4,4,.2))
 curve(dnorm(x)*Nsim*.2,add=T,col="red")
}
```



3.6 Test di Student

Nel caso dello z–test, abbiamo implicitamente supposto di conoscere a priori $\sigma_{\overline{\theta}}$. In caso contrario, la deviazione standard può essere stimata dai dati stessi e la quantità

$$t = \frac{(\overline{\theta} - \theta_0)}{\overline{\sigma_{\overline{\theta}}}} \tag{3.21}$$

non seguirà più una distribuzione gaussiana, poiché $\overline{\sigma_{\overline{\theta}}}$ è, come $\overline{\theta}$, una variabile aleatoria, e il rapporto delle due variabili non segue la distribuzione normale.
Come dimostrato da W.S. Gosset[7], la variabile t segue la distribuzione detta di Student

$$d_t(t) = \frac{\Gamma\left((N+1)/2\right)}{\Gamma\left(N/2\right)\sqrt{\pi N}} \left(1 + \frac{t^2}{N}\right)^{-(N+1)/2} \tag{3.22}$$

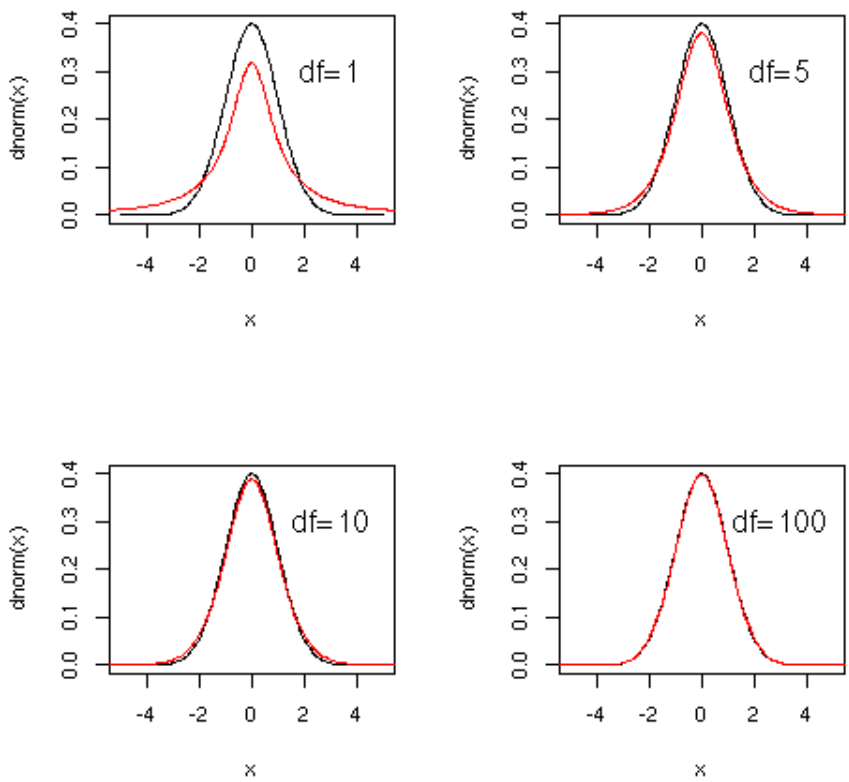
dove N è il numero di gradi di libertà, pari al numero di osservazioni meno 1 (quello utilizzato per stimare $\sigma_{\overline{\theta}}$).
Più in generale si dimostra che se Z è una variabile distribuita secondo la distribuzione normale standard, e χ^2 una variabile distribuita secondo la distribuzione di Pearson con N gradi di libertà, allora la variabile $Z\sqrt{N/\chi^2}$ segue la distribuzione di Student con N gradi di libertà.

Nel software R, la PDF distribuzione di Student è calcolata dalla funzione dt(), e, al solito, le funzioni pt(), qt(), rt() forniscono la funzione di ripartizione, la quantile, e il generatore random.

La distribuzione di Student è importante per campioni a bassa statistica, poiché in pratica differisce dalla gaussiana standard solo per $N \ll 100$.

Esempio 3.6.1 *Confronto fra distribuzioni di Student e di Gauss*
Il seguente codice permette di visualizzare la differenza fra la distribuzione di Student e la distribuzione normale standard in funzione del numero di gradi di libertà.

```
seestudent= function(df=1) {
 curve(dnorm(x),-5,5)
 curve(dt(x,df=df),add=T,col="red")
}
```



Se dunque disponiamo di un campione di bassa statistica e vogliamo testare l'ipotesi $\theta = \theta_0$, dovremo effettuare, anziché uno z–test, un test di Student, o t–test, confrontando il valore osservato di t con la distribuzione di Student attesa nell'ipotesi nulla. Le formule sono le stesse del caso dello z–test in tabella 3.1, dopo aver sostituito z con t, e le distribuzioni della gaussiana standard con quella di Student.

Il t–test è dunque usato per testare se una serie di misure indipendenti $(x_i, i = 1 \ldots N)$ di una variabile normale x sono compatibili con l'avere un dato valore atteso θ_0. In questo caso

$$t = \frac{(\overline{x} - \theta_0)}{\overline{\sigma_{\overline{x}}}} = \frac{(\overline{x} - \theta_0)\sqrt{N}}{\overline{\sigma_x}} \tag{3.23}$$

Il test è spesso eseguito anche per confrontare il valore atteso di due campioni indipendenti $(x_{1i}, i = 1 \dots N_1)$ e $(x_{2i}, i = 1 \dots N_2)$, in cui si possa suppore che la distribuzione sia gaussiana e con la stessa deviazione standard nei due casi. Supponiamo ad esempio di voler testare l'effetto di un sonnifero misurando le ore di sonno su un campione di volontari. Per sincerarsi che l'effetto osservato non dipenda da particolari caratteristiche del nostro gruppo, lo possiamo dividere casualmente in due, somministrando il sonnifero al primo campione e un placebo al "campione di controllo". Nell'ipotesi nulla (il sonnifero non ha effetto) ci aspettiamo che la variabile "ore di sonno" abbia la stessa distribuzione, che assumiamo normale, per i due campioni.

In questo caso la variabile $\Delta = \overline{x_1} - \overline{x_2}$ nell'ipotesi nulla seguirà una distribuzione normale con valore atteso nullo e varianza $\sigma^2_\Delta = \sigma^2/N_1 + \sigma^2/N_2$. La quantità

$$\chi^2_\Delta = \frac{\sum_{i=1}^{N_1}(x_{1i} - \overline{x_1})^2}{\sigma^2} + \frac{\sum_{j=1}^{N_2}(x_{2j} - \overline{x_2})^2}{\sigma^2}$$

è la somma di due variabili χ^2 con $N_1 - 1$ e $N_2 - 1$ gradi di libertà, che sarà dunque una variabile χ^2 con $N_1 + N_2 - 2$ gradi di libertà.
Dunque, in questo caso la variabile di Student è

$$t_\Delta = \frac{\Delta}{\sigma_\Delta}\sqrt{\frac{N_1 + N_2 - 2}{\chi^2_\Delta}} = \Delta\sqrt{\frac{N_1 N_2}{N_1 + N_2}}\sqrt{\frac{N_1 + N_2 - 2}{\sum_{i=1}^{N_1}(x_{1i} - \overline{x_1})^2 + \sum_{j=1}^{N_2}(x_{2j} - \overline{x_2})^2}} \tag{3.24}$$

e segue, nell'ipotesi nulla, la distribuzione di Student con $N_1 + N_2 - 2$ gradi di libertà.

Se invece le misure non sono indipendenti, ma "accoppiate", ovvero si ha $N_1 = N_2$ e c'è una possibile correlazione solo fra le misure in ogni coppia (x_{1i}, x_{2i}), converrà considerare come variabile $\delta = x_1 - x_2$, e testare l'ipotesi $E(\delta) = 0$, eseguendo un t–test con

$$t_\delta = \frac{\overline{\delta}}{\overline{\sigma_{\overline{\delta}}}} = \frac{\overline{\delta}\sqrt{N}}{\overline{\sigma_\delta}} \tag{3.25}$$

Ad esempio, nell'esempio del test del sonnifero ci converrà misurare le ore di sonno degli stessi individui con e senza sonnifero.

Esempio 3.6.2 *Il classico test sull'effetto dello hyoscyamine hydrobromide*
In questo esempio dovuto allo stesso Gosset, i dati (nel dataframe *sleep*) rappresentano la reazione, in ore di sonno addizionali, di 10 pazienti a due diversi sonniferi (Dextro– e Laevo–hyoscyamine hydrobromide). Si vuole testare nei due casi l'ipotesi che il sonnifero non abbia alcun effetto, determinando il *p–value*. Si vuole inoltre testare se l'effetto dei due sonniferi sia lo stesso.

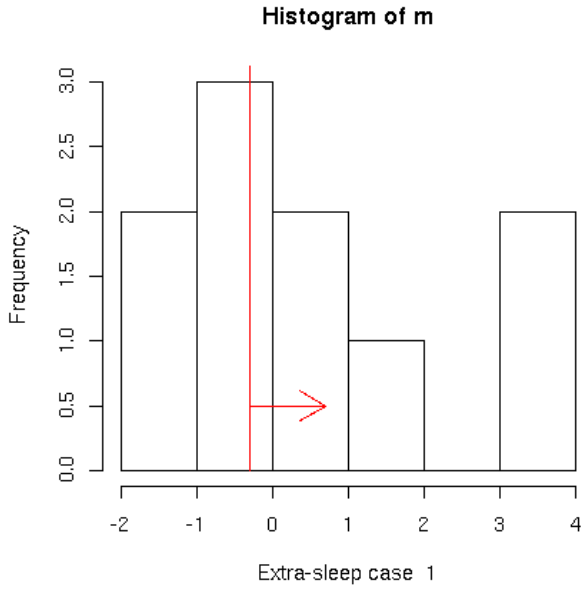
Assumendo che la risposta dei pazienti segua una distribuzione normale con media e deviazione standard incognite, la variabile empirica

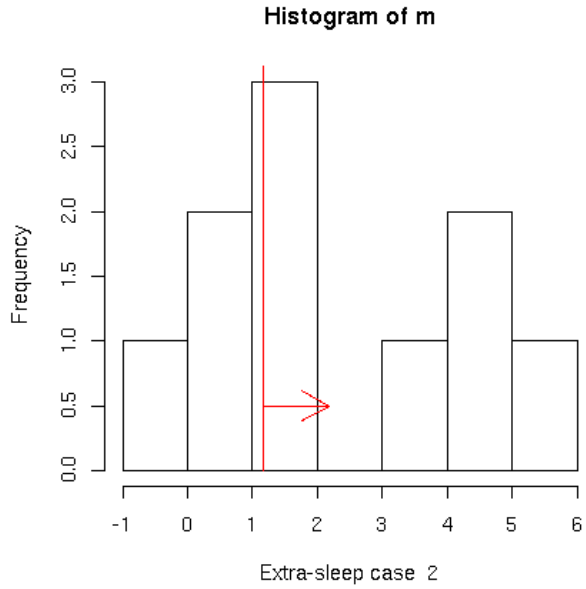
$$t = \frac{(\overline{x} - \lambda_0)}{\overline{\sigma_{\overline{x}}}}$$

(dove $\overline{x}$ e $\overline{\sigma_{\overline{x}}}$ sono gli stimatori della media e del suo errore, e $\lambda_0 = 0$ è il valore atteso nell'ipotesi nulla)
seguirà una distribuzione di Student con $10 - 1$ gradi di libertà.

Assumendo che l'ipotesi alternativa consista in un incremento significativo delle ore di sonno, possiamo eseguire il test di Student ad una coda calcolando dunque il *p–value* come la probabilità di ottenere un valore di t superiore o uguale a quello misurato.
Il seguente codice, oltre ad eseguire il test calcolando il *p–value*, calcola il limite inferiore per il valore atteso dell'effetto del sonnifero, e mostra il risultato graficamente.

```
ttestsleep = function(gr=1,conflevel=0.95) {
 m=subset(sleep,group==gr)$extra
 hist(m,xlab=paste("Extra-sleep case ",gr))
 n=length(m)
 ngl=n-1
 dmean=sd(m)/sqrt(n)
 t=mean(m)/dmean
 cat(" test di compatibilità con 0 (1 coda): pvalue=",(1-pt(t,df=ngl)),"\n")
 nsigma=qt((1+conflevel)/2,df=ngl)
 limite=mean(m)-dmean*qt(conflevel,df=ngl)
 cat("  limite: >",limite," per livello di confidenza=",conflevel,"\n")
 lines(c(limite,limite),c(0,10),col="red")
 arrows(limite,0.5,limite+1,0.5,col="red")
}
> ttestsleep(1)
test di compatibilità con 0 (1 coda): pvalue= 0.1087989
  limite: > -0.2870553  per livello di confidenza= 0.95

> ttestsleep(2)
test di compatibilità con 0 (1 coda): pvalue= 0.002538066
  limite: > 1.169334  per livello di confidenza= 0.95
```





Possiamo dunque concludere che i dati mostrano un effetto significativo solo per il secondo sonnifero.
Il calcolo effettuato è riprodotto dalla funzione di R *t.test()*:

```
> t.test(subset(sleep,group==1)$extra,alternative="greater")

        One Sample t-test

data:  m
t = 1.3257, df = 9, p-value = 0.1088
alternative hypothesis: true mean is greater than 0
95 percent confidence interval:
 -0.2870553        Inf
sample estimates:
mean of x
     0.75

>  t.test(subset(sleep,group==2)$extra,alternative="greater")

        One Sample t-test

data:  subset(sleep, group == 2)$extra
t = 3.6799, df = 9, p-value = 0.002538
alternative hypothesis: true mean is greater than 0
95 percent confidence interval:
 1.169334      Inf
sample estimates:
mean of x
     2.33
```

Se più prudentemente volessimo cosiderare anche l'ipotesi alternativa che il sonnifero abbia un effetto negativo sulle ore di sonno, dovremmo fare un test a due code:

```
ttestsleep2 = function(gr=1,conflevel=0.95) {
 m=subset(sleep,group==gr)$extra
 n=length(m)
 ngl=n-1
 dmean=sd(m)/sqrt(n)
 t=mean(m)/dmean
 cat(" test di compatibilità con 0 (2-code): pvalue=",2*(1-pt(abs(t),df=ngl)),"\n")
 nsigma=qt((1+conflevel)/2,df=ngl)
 cat(" intervallo per livello di confidenza=",conflevel," :", mean(m),
" +/- ",dmean*nsigma,"\n")
}
> ttestsleep2(1)
 test di compatibilità con 0 (2-code): pvalue= 0.2175978
 intervallo per livello di confidenza= 0.95  : 0.75  +/-  1.279780
```

equivalente a

```
>  t.test(subset(sleep,group==1)$extra,alternative="two.sided")
```

```
        One Sample t-test

data:  subset(sleep, group == 1)$extra
t = 1.3257, df = 9, p-value = 0.2176
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 -0.5297804  2.0297804
sample estimates:
mean of x
     0.75
```

Infine, eseguiamo un test sulla differenza di effetto δ su ogni paziente:

```
> delta = subset(sleep,group==2)$extra - subset(sleep,group==1)$extra
> t.test(delta,alternative="two.sided",conf.level=0.95)

        One Sample t-test

data:  delta
t = 4.0621, df = 9, p-value = 0.002833
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 0.7001142 2.4598858
sample estimates:
mean of x
     1.58
```

concludendo che l'ipotesi di uguale effetto ha un *p–value* pari allo 0.3% e può dunque essere esclusa con una confidenza del 99.7%.

3.7 Test di Fisher

Abbiamo visto come il test di Student permetta di testare l'ipotesi che due campioni normali abbiano lo stesso valore atteso, assumendo uguale varianza. Vogliamo ora testare l'ipotesi di uguale varianza, considerando il rapporto fra le stime della varianza dei due campioni:

$$F = \frac{\overline{\sigma_1}}{\overline{\sigma_2}} = \frac{\sum_{i=1}^{N_1}(x_{1i} - \overline{x_1})^2}{(N_1 - 1)} \frac{(N_2 - 1)}{\sum_{i=1}^{N_2}(x_{2i} - \overline{x_2})^2} \tag{3.26}$$

Nell'ipotesi nulla $\sigma_1^2 = \sigma_2^2 = \sigma_2$, ci aspettiamo che $F \sim 1$, e possiamo riscriverlo come

$$F = \frac{\sum_{i=1}^{N_1}(x_{1i} - \overline{x_1})^2}{\sigma^2(N_1 - 1)} \frac{\sigma^2(N_2 - 1)}{\sum_{i=1}^{N_2}(x_{2i} - \overline{x_2})^2} = \frac{\chi_1^2}{(N_1 - 1)} \frac{(N_2 - 1)}{\chi_2^2} \tag{3.27}$$

dove, nell'ipotesi di normalità, le variabili χ_j^2 seguono la distribuzione di Pearson con $f_j = N_j - 1$ gradi di libertà, ed F risulta essere il rapporto di due χ^2 normalizzati. Si dimostra che in queste ipotesi F segue la **distribuzione F di Fisher**

$$d_F = \left(\frac{f_1}{f_2}\right)^{f_1/2} \frac{\Gamma\left((f_1+f_2)/2\right)}{\Gamma(f_1/2)\Gamma(f_2/2)} F^{f_1/2-1} \left(1 + \frac{f_1}{f_2}F\right)^{-(f_1+f_2)/2} \tag{3.28}$$

ed ha valore atteso

$$E(F) = \frac{f_2}{(f_2-2)} \qquad (\text{per } f_2 > 2) \tag{3.29}$$

e varianza

$$\frac{2f_2^2(f_1+f_2-2)}{f_1(f_2-2)^2(f_2-4)} \qquad (\text{per } f_2 > 4) \tag{3.30}$$

Nel software R, la funzione df() implementa la funzione F di Fisher.
Se l'ipotesi alternativa è $\sigma_1 > \sigma_2$, il test di Fisher, o **F–test**, sarà il test a una coda

$$F < q_F(1-\alpha_s; f_1, f_2) \tag{3.31}$$

Nel caso di test a due code

$$q_F(\alpha_s/2; f_1, f_2) < F < q_F(1-\alpha_s/2; f_1, f_2) \tag{3.32}$$

che per simmetria è equivalente a

$$\begin{aligned} F &< q_F(1-\alpha_s/2; f_1, f_2) \\ 1/F &< q_F(1-\alpha_s/2; f_2, f_1) \end{aligned} \tag{3.33}$$

3.8 Analisi della varianza

La più importante applicazione del test di Fisher consiste nell'**analisi della varianza**, in cui si testa se n_g campioni sono compatibili con avere lo stesso valore atteso, nell'ipotesi di normalità con σ fissata ma non nota. Il test è utile in particolare per sapere se il valore atteso di una variabile x, misurata con un certo errore costante σ, può dipendere da una variabile categoriale in base alla quale è possibile suddividere in gruppi il nostro campione.

Indichiamo con x_{ij} i nostri dati dove $i = 1, \ldots n_g$ è l'indice di gruppo, $j = 1, \ldots N_i$ è l'indice all'interno del gruppo, $N = \sum_{i=1}^{n_g} N_i$ è il numero totale di valori. Il valor medio di ciascun gruppo è

$$\overline{x_i} = \frac{1}{N_i} \sum_{j=1}^{N_i} x_{ij} \tag{3.34}$$

e il valor medio complessivo è

$$\overline{x} = \frac{1}{N} \sum_{i=1}^{n_g} \sum_{j=1}^{N_i} x_{ij} = \frac{1}{N} \sum_{i=1}^{n_g} N_i \overline{x_i} \tag{3.35}$$

La somma degli scarti quadratici è

$$
\begin{aligned}
Q &= \sum_{i=1}^{n_g}\sum_{j=1}^{N_i}(x_{ij}-\overline{x})^2 = \\
&= \sum_{i=1}^{n_g}\sum_{j=1}^{N_i}\left[(x_{ij}-\overline{x_i})+(\overline{x_i}-\overline{x})\right]^2 = \\
&= \sum_{i=1}^{n_g}\sum_{j=1}^{N_i}(x_{ij}-\overline{x_i})^2 \;+\; \sum_{i=1}^{n_g}\sum_{j=1}^{N_i}(\overline{x_i}-\overline{x})^2 \;+\; 2\sum_{i=1}^{n_g}\sum_{j=1}^{N_i}(x_{ij}-\overline{x_i})(\overline{x_i}-\overline{x})
\end{aligned}
\tag{3.36}
$$

Poichè

$$
\sum_{i=1}^{n_g}\sum_{j=1}^{N_i}(x_{ij}-\overline{x_i})(\overline{x_i}-\overline{x}) = \sum_{i=1}^{n_g}(\overline{x_i}-\overline{x})\sum_{j=1}^{N_i}(x_{ij}-\overline{x_i}) = 0 \tag{3.37}
$$

possiamo separare Q in un termine che dipende dagli scarti all'interno di ciascun gruppo

$$
Q_W \equiv \sum_{i=1}^{n_g}\sum_{j=1}^{N_i}(x_{ij}-\overline{x_i})^2 \tag{3.38}
$$

ed uno che dipende dalle differenze fra i gruppi

$$
Q_B \equiv \sum_{i=1}^{n_g} N_i(\overline{x_i}-\overline{x})^2 \tag{3.39}
$$

Nell'ipotesi nulla, ci aspettiamo che Q/σ^2 segua la distribuzione χ^2 con $N-1$ gradi di libertà. Inoltre, le variabili $\overline{x_i}$ sono anch'esse variabili normali con valore atteso pari a $E(x)$ e varianza pari a σ^2/N_i, e dunque Q_B/σ^2 dovrà seguire la distribuzione χ^2 con (n_g-1) gradi di libertà. Di conseguenza $Q_W/\sigma^2 = (Q-Q_B)/\sigma^2$ seguirà la distribuzione χ^2 con $(N-n_g)$ gradi di libertà. Si ottiene dunque che, nell'ipotesi nulla, il rapporto

$$
F = \frac{Q_B}{\sigma^2(n_g-1)}\frac{\sigma^2(N-n_g)}{Q_W} = \frac{Q_B}{(n_g-1)}\frac{(N-n_g)}{Q_W} \tag{3.40}
$$

segue la distribuzione di Fisher con $f_1 = (n_g-1)$, $f_2 = (N-n_g)$. Poichè qualunque ipotesi alternativa produce valori di Q_B, e dunque di F, più alti, l'analisi della varianza consiste in un F–test a una coda (cfr eq. 3.31), con

$$
p\text{–}value = 1 - p_F(F; n_g-1, N-n_g) \tag{3.41}
$$

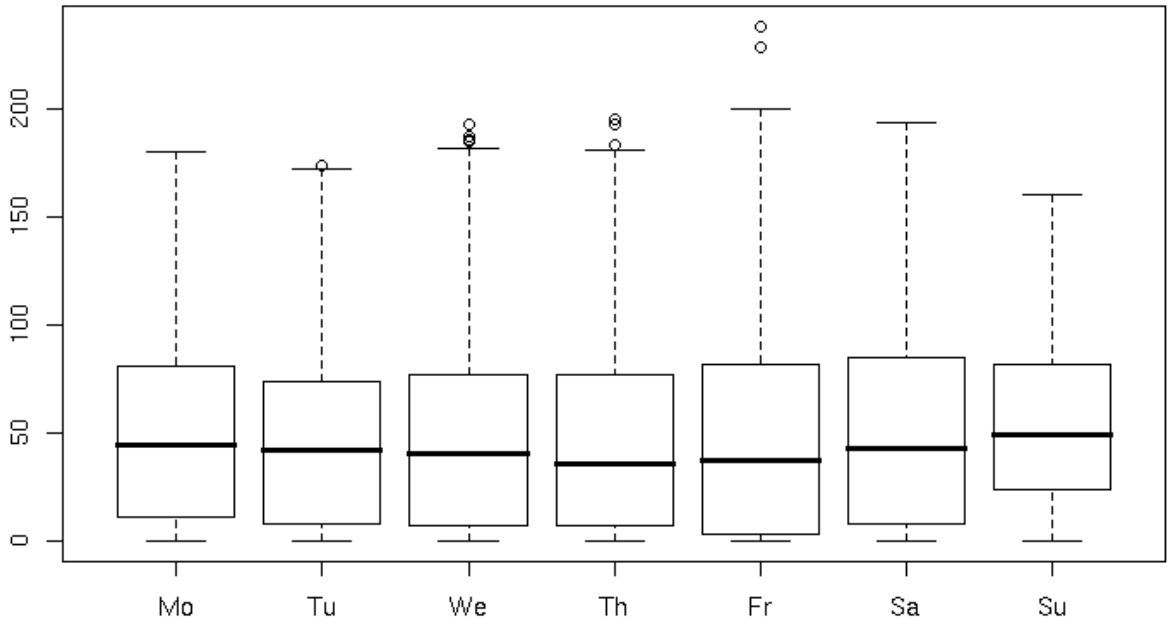
L'analisi della varianza può essere vista come una estensione al caso $n_g > 2$ del test di Student dell'eq. 3.24. Nel caso $n_g = 2$ i due tests sono del tutto equivalenti, poiché si dimostra che $F = t_\Delta^2$ segue una distribuzione di Fisher con $f_1 = 1$, $f_2 = N-2$.

Esempio 3.8.1 *Misure di inquinamento al giardino di Boboli*
I dati nella tabella del file
/afs/math.unifi.it/service/Rdsets/ozono_boboli_2005.rdata

contengono i valori di concentrazione di ozono (in μg/m^3), rivelati nel giardino di Boboli a Firenze con cadenza oraria nel corso del 2005. Ci chiediamo se ci sia una dipendenza dal giorno della settimana.

Cominciamo col visualizzare i valori salienti della distribuzione dei valori per ogni giorno della settimana, utilizzando la funzione *boxplot()*:

```
boboli=read.table("/afs/math.unifi.it/service/Rdsets/ozono_boboli_2005.rdata")
#ordiniamo i giorni della settimana
giorno = ordered(boboli$wday,levels=c("Mo","Tu","We","Th","Fr","Sa","Su"))
boxplot(boboli$level ~ giorno)
```



Non essendo evidente dal grafico la risposta al nostro quesito, eseguiamo un'analisi della varianza (si noti l'uso della funzione *tapply()* per il calcolo della media di ciascun gruppo):

```
boboli.weekday = function() {
 ni=tapply(boboli$level, giorno, length)
 ml=mean(boboli$level)
 mi= tapply(boboli$level, giorno, mean)
 Qb= sum(ni*(mi-ml)^2)
 Qw= sum((boboli$level - mi[giorno])^2)
 ng = nlevels(giorno)
```

```
 n = nrow(boboli)
 F = Qb / (ng - 1) /(Qw/ (n-ng) )
 cat("F=", F," p-value=",1-pf(F,df1=ng-1, df2=n-ng),"\n")
}
> boboli.weekday()
F= 6.120735  p-value= 2.054779e-06
```

Escludiamo dunque l'ipotesi nulla e concludiamo che il livello di ozono dipende dal giorno della settimana.
Il calcolo può essere eseguito più rapidamente, utilizzando l'apposita funzione *aov()* di R:

```
> summary(aov(level ~ wday, data = boboli))
              Df   Sum Sq  Mean Sq F value    Pr(>F)
wday           6    63883    10647  6.1207 2.055e-06 ***
Residuals   8753 15226156     1740
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

3.9 Il teorema di Neyman–Pearson

Abbiamo visto nei nostri esempi come diversi tests possano essere effettuati per affrontare uno stesso problema. In generale il test preferibile è quello più potente. Un test è detto "uniformemente più potente" se si può dimostrare che è il più potente indipendentemente dall'ipotesi alternativa. Questo non è sempre possibile, ed in generale la scelta del test più potente può dipendere dall'ipotesi alternativa.
Il teorema di Neyman–Pearson[1] ci fornisce il criterio per determinare se un test è il più potente possibile per una data ipotesi nulla H_0 ed una data ipotesi alternativa H_1, a parità di α_s . Il teorema afferma che la statistica di test ottimale è il rapporto r fra le funzioni di likelihood nelle due ipotesi, detto *likelihood ratio*. La regione accettata del test più potente si costruisce dunque tramite la condizione

$$r \equiv \frac{\mathcal{L}(\mathcal{S}|H_0)}{\mathcal{L}(\mathcal{S}|H_1)} > c(\alpha_s) \tag{3.42}$$

dove $c(\alpha_s)$ è una costante da fissare in base alla significatività voluta.

Ad esempio, consideriamo il problema del test del valore atteso di una variabile normale, con σ fissata, considerato nei paragrafi 3.4 e 3.6. Il criterio di Neyman–Pearson ci suggerisce il test

$$\frac{\prod_i \phi_G(x_i|\theta_0)}{\prod_i \phi_G(x_i|\theta_1)} = exp\left(\frac{-\sum_i (x_i-\theta_0)^2 + \sum_i (x_i-\theta_1)^2}{2\sigma^2}\right) > c \tag{3.43}$$

[1] si veda [1] per la dimostrazione

ovvero, passando a $\log r$ ed eliminando i termini che non dipendono da θ e x_i

$$2\sum_i x_i(\theta_0-\theta_1) - N(\theta_0^2-\theta_1^2) > c' \implies (\theta_0-\theta_1)\left(\overline{x} - \frac{(\theta_0+\theta_1)}{2}\right) > c'' \tag{3.44}$$

Si ottiene dunque che:

- la statistica di test da usare è $\overline{x}$;
- fissato θ_1, il test più potente è lo z–test a una coda, scegliendo la coda destra o sinistra in base al segno di $(\theta_0-\theta_1)$;
- se il segno di $(\theta_0-\theta_1)$ non è fissato, non esiste un test uniformemente più potente.

3.10 Ipotesi multiple: problemi di classificazione

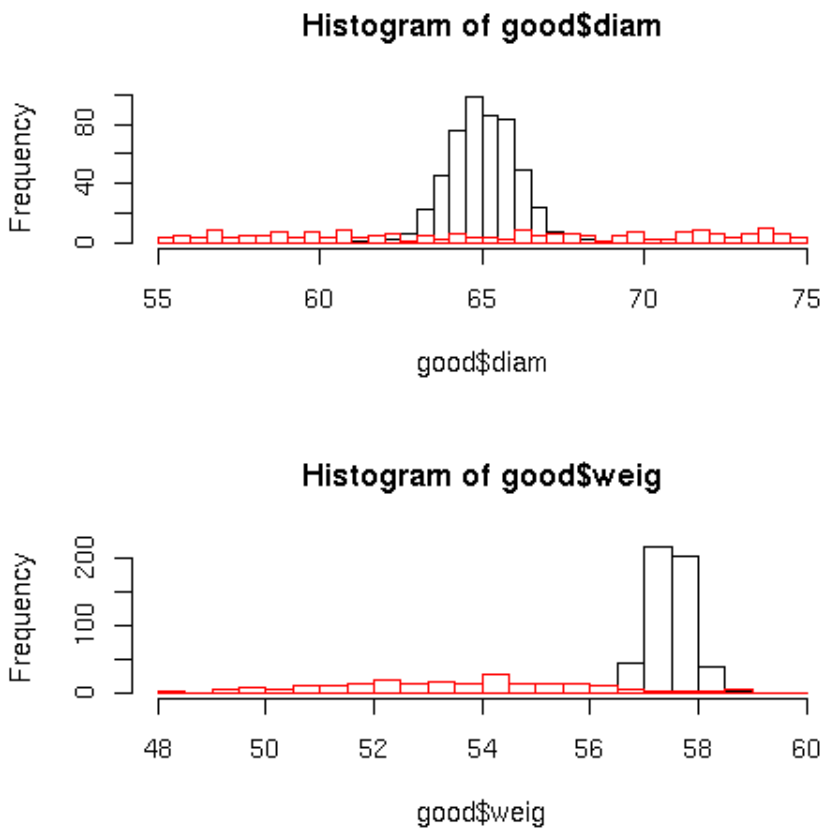
Consideriamo ora un campione di dati in cui ogni elemento è generato da una PDF $\mathcal{d}(x|H_i)$. Conoscendo le n_h possibili ipotesi H_i, vorremmo classificare i nostri dati separando gli elementi in base all'ipotesi più plausibile per ciascuno. Un esempio può essere la classificazione di un campione di piante secondo la specie di appartenenza, oppure un algoritmo per distinguere un aereo civile o militare sulla base delle osservazioni di un radar.
Faremo dunque un test ad ipotesi multiple per ciascun elemento del campione; il criterio ottimale per separare ciascuna coppia di ipotesi è dato dal teorema di Neyman–Pearson.

In molti problemi pratici le ipotesi H_i non sono completamente specificate a priori, ma dipendono da parametri che devono essere ottenuti empiricamente dai dati stessi. In questi casi è necessario disporre di un *campione di training* per stimare le $\mathcal{d}(x|H_i)$, che dovrà essere indipendente dal campione su cui effettuare l'analisi di classificazione.

Esempio 3.10.1 *Un esempio di controllo qualità*
I dati tabulati nel file
/afs/math.unifi.it/service/Rdsets/tennisballs.rdata
rappresentano i valori di diametro e peso di palline da tennis, misurati in modo automatico dalla catena di montaggio che le produce.
Si assume che, per le palline prodotte correttamente, le due variabili seguano una distribuzione gaussiana centrata intorno al valore nominale (65 mm di diametro, 57.5 g di peso). A causa di un difetto di produzione, alcune palline non vengono correttamente pressate e risultano avere una forma irregolare. Assumiamo che il diametro misurato delle palline "sbagliate" abbia una distribuzione uniforme nell'intervallo fra 55 e 75 mm, mentre il peso segua una distribuzione gaussiana. Consideriamo le due variabili indipendenti in entrambi i casi. Vogliamo ottenere il miglior criterio di controllo di qualità automatico, che si deve basare unicamente sui valori di diametro e peso, sulla base di un campione di training (le palle controllate manualmente che hanno un valore definito della variabile *clas*), e stimare la purezza del campione che supera il criterio di qualità.

Cominciamo col confrontare visivamente la distribuzione delle due variabili utilizzando il campione di training

```
tb=read.table('/afs/math.unifi.it/service/Rdsets/tennisballs.rdata')
training = tb[is.na(tb$clas)==0,]
data = tb[is.na(tb$clas)==1,]
good=training[training$clas=='OK',]
bad =training[training$clas=='BAD',]
par(mfrow=c(2,1))
hist(good$diam,breaks=seq(55,75,0.5))
hist(bad$diam,breaks=seq(55,75,0.5),add=T,border="red")
hist(good$weig,breaks=seq(46,60,0.5))
hist(bad$weig,breaks=seq(46,60,0.5),add=T,border="red")
par(mfrow=c(1,1))
```



Verifichiamo anche che non ci siano correlazioni evidenti fra le variabili, in modo da poter assumere, per entrambe le classi "good" e "bad":

$$d(w,d) = d_w(w)d_d(d)$$

```
seecor = function() {
 cor.good = cor(good[,1:2])
 dcor.good = (1-cor.good^2) / sqrt(nrow(good))
 cor.bad = cor(bad[,1:2])
 dcor.bad = (1-cor.bad^2) / sqrt(nrow(bad))
 cat("Matrice di correlazione GOOD:","\n")
 print(cor.good)
```

```
 cat(" +/- ","\n")
 print(dcor.good)
 cat("\nMatrice di correlazione BAD:","\n")
 print(cor.bad)
 cat(" +/- ","\n")
 print(dcor.bad)
}
> seecor()
Matrice di correlazione GOOD:
            diam        weig
diam 1.000000000 0.002842401
weig 0.002842401 1.000000000
 +/-
           diam       weig
diam 0.00000000 0.01924485
weig 0.01924485 0.00000000

Matrice di correlazione BAD:
           diam       weig
diam 1.00000000 0.06071539
weig 0.06071539 1.00000000
 +/-
           diam       weig
diam 0.00000000 0.03003999
weig 0.03003999 0.00000000
```

Per stimare i parametri incogniti delle distribuzioni per i due campioni "good" e "bad", utilizziamo una metà del campione di training, scegliendo a caso la metà dei valori

```
nt = nrow(training)
set.seed(99)
acaso=sample(1:nt,as.integer(nt/2))
tr1 =training[acaso,]
tr2 =training[-acaso,]
# usiamo il primo campione per stimare i parametri del modello
mu.d.good = 65
sigma.d.good = sd(tr1[tr1$clas == 'OK',]$diam)
mu.w.good = 57.5
sigma.w.good = sd(tr1[tr1$clas == 'OK',]$weig)
min.d.bad = 55
max.d.bad = 75
mu.w.bad = mean(tr1[tr1$clas == 'BAD',]$wei)
sigma.w.bad = sd(tr1[tr1$clas == 'BAD',]$wei)
```

(si noti l'uso della funzione *sample()* per estrarre casualmente metà delle righe dal dataframe, e della funzione *set.seed()* per poter riprodurre la stessa sequenza pseudo–random nel caso di ripetuta esecuzione del codice)

Il test più potente per separare i due campioni è ottenuto con un taglio sul rapporto di likelihood

```
tennis.loglikratio= function(d,w) {
 log( (dnorm(d,mean=mu.d.good,sd=sigma.d.good)*
       dnorm(w,mean=mu.w.good,sd=sigma.w.good)) /
      (dunif(d,min=min.d.bad,max=max.d.bad) *
       dnorm(w,mean=mu.w.bad,sd=sigma.w.bad) ))
}
```

Il criterio di qualità sarà dunque definito da $tennis.loglikratio(d, w) > k$ dove la costante k dipende dalla significatività/potenza che vogliamo ottenere. Una scelta naturale per k è il valore 0 ($r(good/bad) > 1$).
Per valutare la purezza delle N' palline che superano il taglio

$$p = \frac{N'_{good}}{N'_{good} + N'_{bad}} = \frac{(1-\alpha_s)N_{good}}{(1-\alpha_s)N_{good} + \beta N_{bad}} = \frac{1}{1 + \frac{\beta}{(1-\alpha)}\frac{N_{bad}}{N_{good}}}$$

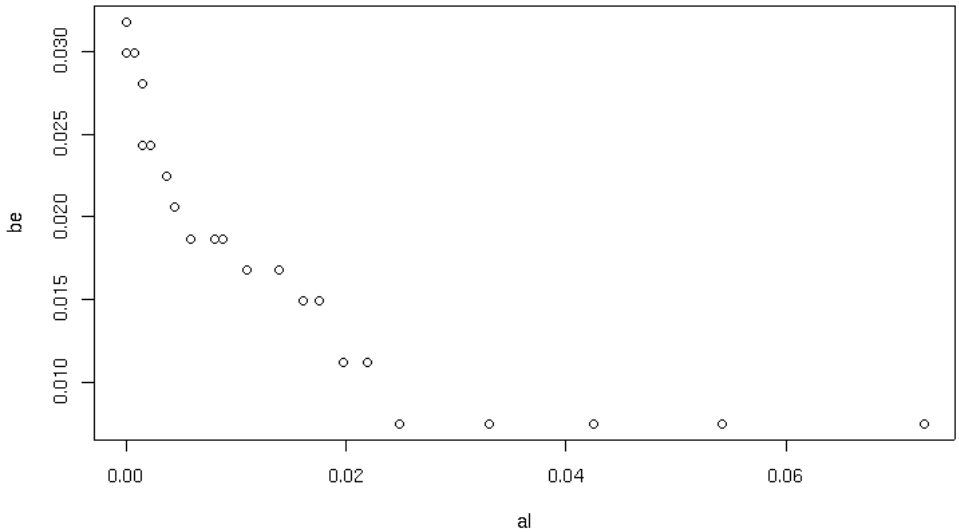
è necessario stimare, utilizzando la seconda metà del campione di training, le probabilità di errore di primo e secondo tipo α_s e β:

```
tr2.good= tr2[tr2$clas == 'OK',]
tr2.bad= tr2[tr2$clas == 'BAD',]
ll.good = tennis.loglikratio(tr2.good$diam,tr2.good$weig)
ll.bad  = tennis.loglikratio(tr2.bad$diam ,tr2.bad$weig)

ll.data =  tennis.loglikratio(data$diam,data$weig)

computeAlfaBeta = function(cut = 0, speak=T) {
  nok = nrow(tr2.good)
  alpha = nrow(tr2.good[ll.good<cut,])/ nok
  dalpha=sqrt(alpha*(1-alpha)/nok)

  nbad = nrow(tr2.bad)
  beta = nrow(tr2.bad[ll.bad >=cut ,])/ nbad
  dbeta= sqrt(beta*(1-beta)/nbad)

  if(speak) {
   cat ("alpha=",alpha, " +/- ",dalpha,"\n")
   cat ("beta=",beta, " +/- ",dbeta,"\n")
  }

  data.ratio =  nrow(data[ ll.data < cut,]) /  nrow(data[ ll.data >= cut,])

  p = (1-alpha) /(1-alpha + beta * data.ratio)
  sigma.p = sqrt( ((1-alpha)* dalpha)^2 + (beta*data.ratio*dbeta)^2  ) /
        (1-alpha+beta*data.ratio)^2
  if (speak) {
```

```
   cat ("purezza=",p, " +/- ",sigma.p,"\n")
  }
  c(alpha,beta,p)
}
> computeAlfaBeta()
alpha= 0.004395604  +/-  0.001790550
beta= 0.02056075  +/-  0.006135237
purezza= 0.9991238  +/-  0.001795313
```

Per $k = 0$, otteniamo dunque $p = 0.999 \pm 0.002$ (si noti che l' "errore standard" in questo caso dovrebbe essere sostituito con un calcolo più accurato dell'intervallo di confidenza).
Per scegliere il criterio di qualità più opportuno in base alle nostre esigenze, è utile fare un grafico di α_s in funzione di β al variare del taglio. Aumentando il taglio k, aumenterà la potenza $(1 - \beta)$, e dunque la purezza, ma anche la frazione α_s di palline buone che vengono scartate.

```
plotPerformances = function() {
 al=c()
 be=c()
 pu=c()
 i=1
 cuts=seq(-3,3,0.2)
 for (cut in cuts ) {
  perf=computeAlfaBeta(cut,speak=0)
  al[i]= perf[1]
  be[i]= perf[2]
  pu[i]= perf[3]
  i=i+1
 }
 plot(al,be)
}
```



3.10.1 Discriminante lineare di Fisher

Nei casi in cui abbiamo a disposizione n variabili $\underline{x}$ per classificare il nostro campione, dobbiamo conoscere la funzione densità n–dimensionale $d(\underline{x}|H_j)$ per costruire il rapporto r. Spesso questo risulta essere troppo complesso in pratica, a meno che le variabili non siano indipendenti e la funzione possa essere fattorizzata, come nell'esempio 3.10.1:

$$d(\mathcal{S}, d|H_j) = \prod_{i=1}^{n} d_i(x_i|H_j) \tag{3.45}$$

Per un generico campione con variabili correlate possiamo comunque, nell'ipotesi in cui le correlazioni possano essere approssimate a funzioni lineari, applicare una trasformazione lineare in modo da ottenere un nuovo set di variabili indipendenti $\underline{y}$. Questa operazione, detta **analisi delle componenti principali** (ACP) consiste semplicemente nel diagonalizzare la matrice di correlazione R:

$$URU^T = \Lambda \tag{3.46}$$

$$\underline{y} = U\underline{x'} \tag{3.47}$$

dove $\underline{x'}$ è il vettore delle variabili ridotte

$$x'_i = \frac{x_i - E(x_i)}{\sigma(x_i)} \tag{3.48}$$

Λ è la matrice diagonale degli autovalori $\lambda_1 = \sigma^2(y_i)$ e U è una matrice ortogonale, essendo R simmetrica. Si noti che è raccomandabile diagonalizzare R anziché la matrice varianza–covarianza, per rendere il risultato indipendente da trasformazioni di scala delle variabili.
Questo procedimento è spesso usato nell'analisi esplorativa di un campione per ridurre il set di variabili a quelle linearmente indipendenti. Infatti, una variabile ridondante può essere individuata da un autovalore molto minore rispetto agli altri. Nei problemi di classificazione bisogna comunque valutare con attenzione se un autovalore piccolo è dovuto alla effettiva ridondanza della variabile, o ad una caratteristica saliente del campione che potrebbe, al contrario, essere essenziale per la potenza del test di classificazione.

L'analisi delle componenti principali ci permette, almeno in un'approssimazione al primo ordine, di fattorizzare le funzioni densità e modellizzare più facilmente il rapporto di likelihood. L'**analisi discriminante di Fisher** fornisce un metodo più diretto, parente stretto della ACP, per derivare la statistica di test opportuna a separare due ipotesi H_0 e H_1. Si tratta di prendere la combinazione lineare t, detta **discriminante di Fisher**

$$t = \sum_i a_i x_i = \underline{a} \cdot \underline{x} \tag{3.49}$$

che massimizzi la quantità

$$J = \frac{[E(t|H_0) - E(t|H_1)]^2}{[\sigma^2(t|H_0) + \sigma^2(t|H_1)]^2} \tag{3.50}$$

L'idea è quella di proiettare i nostri dati sulla direzione nel piano n–dimensionale che massimizza la distanza fra i baricentri delle distribuzioni per le due ipotesi, normalizzata con la propria varianza.
La soluzione è

$$\underline{a} = k\left[V(\underline{x}|H_0) + V(\underline{x}|H_1)\right]^{-1}\left[E(\underline{x}|H_0) - E(\underline{x}|H_1)\right] \tag{3.51}$$

dove k è una costante arbitraria e V è la matrice varianza–covarianza.

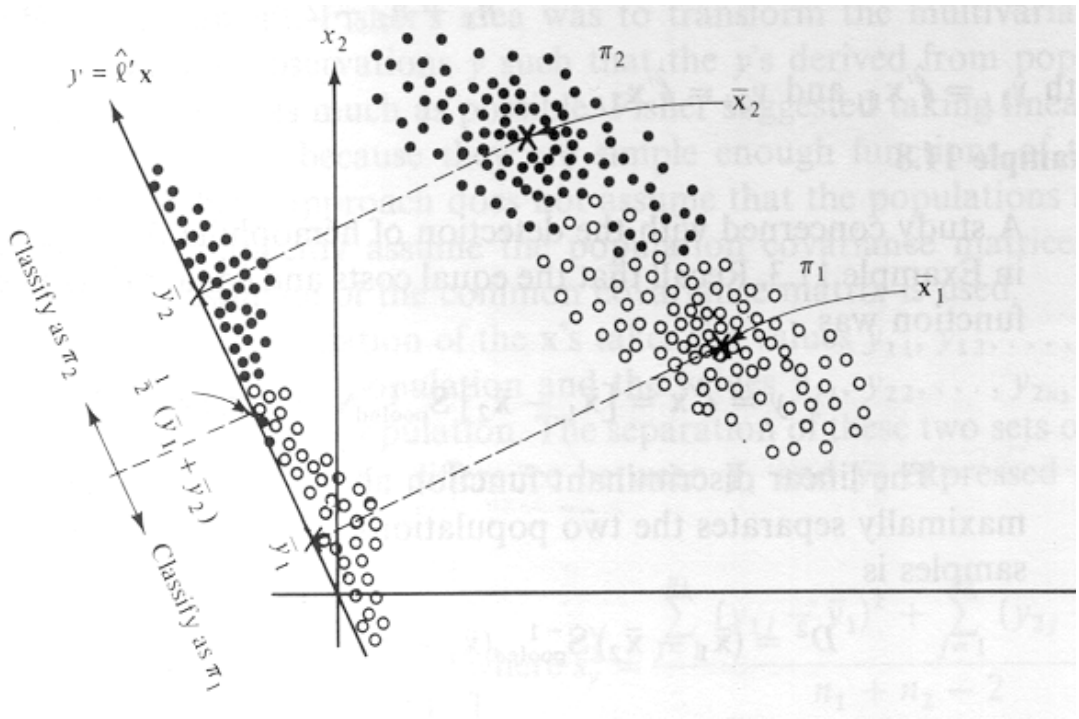


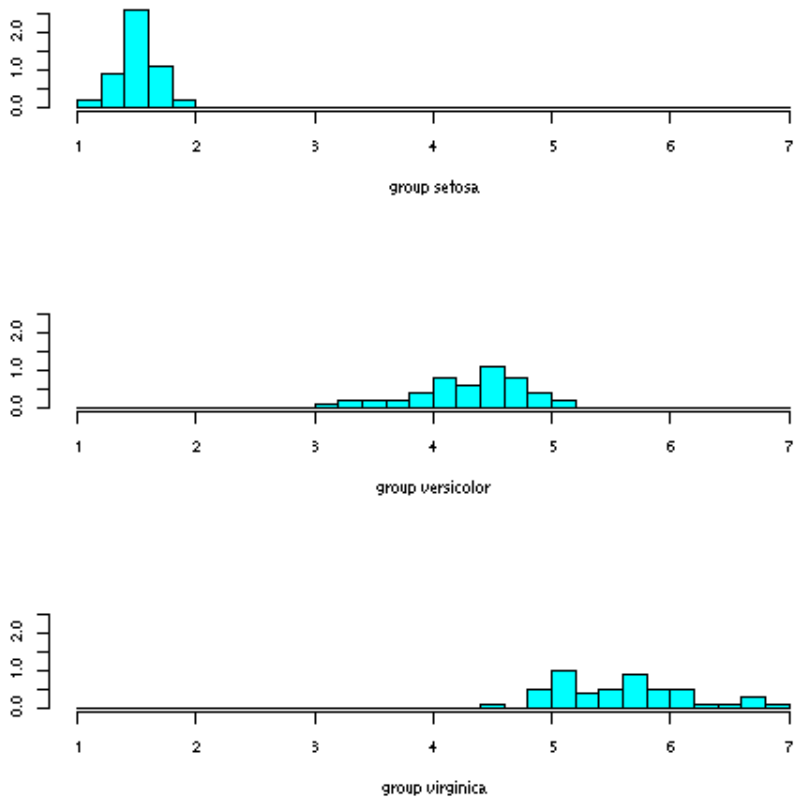
Figura 3.1: Rappresentazione grafica dell'analisi discriminante di Fisher

Si dimostra che se le variabili seguono una distribuzione gaussiana multivariata e nelle due ipotesi hanno la stessa matrice di correlazione, l'analisi di Fisher coincide con il test più potente, essendo $r \propto e^t$.

Esempio 3.10.2 *Il classico problema di classificazione dell' iris*
Il dataframe iris contiene i valori di quattro parametri morfologici per 150 fiori di iris di tre diverse specie. Vogliamo ottenere il miglior discriminatore lineare, sulla base delle 4 variabili misurate, per separare la specie "versicolor" dalle altre. Determineremo il criterio di separazione utilizzando la metà del campione (selezionandola casualmente) e usando l'altra metà per valutare graficamente la separazione ottenuta.

Da un controllo visuale delle 4 variabili risulta evidente che la specie "setosa" può essere isolata dalle altre con un semplice taglio sulla variabile Petal.Length

```
library(MASS)
ldahist(iris$Petal.Length, iris$Species)
```

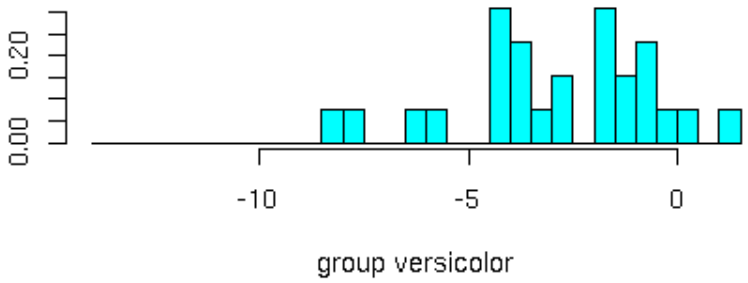


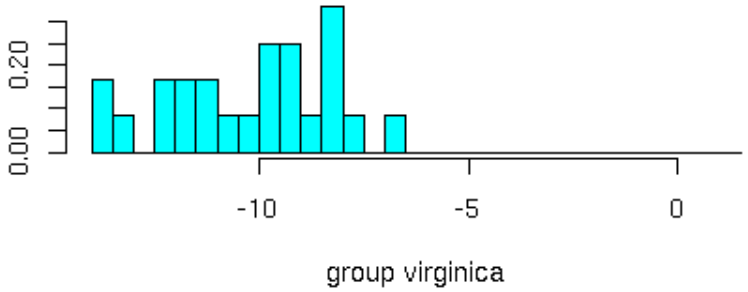
Costruiamo dunque un discriminante lineare per separare l'ipotesi "versicolor" dall'ipotesi alternativa "virginica":

```
myiris=iris[iris$Petal.Length>2.5,]
set.seed(99)
samp=sample(1:100,50)
myiris.train=myiris[samp,]
myiris.test =myiris[-samp,]

# calcolo delle medie nelle due ipotesi
mu0 = mean(myiris.train[myiris.train$Species=="versicolor",1:4])
mu1 = mean(myiris.train[myiris.train$Species=="virginica",1:4])

#calcolo delle matirice var/cov nelle due ipotesi
v0 = var(myiris.train[myiris.train$Species=="versicolor",1:4])
v1 = var(myiris.train[myiris.train$Species=="virginica",1:4])

#calcolo dei coefficienti lineari
a = solve(v0+v1) %*% (mu0-mu1)
print(a)
```

Possiamo a questo punto calcolare la funzione discriminante e osservare la separazione ottenuta sul campione di test:

```
discr = as.matrix(myiris.test[1:4]) %*% a
myiris.test = data.frame(myiris.test,fisher=discr)
ldahist(myiris.test$fisher,myiris.test$Species)
```





Il discriminante può essere ottenuto in modo equivalente con la funzione di R *lda()*:

```
iris.lda2=lda(Species ~ . , myiris.train,prior=c(0,1,1)/2)
print(iris.lda2$scaling)
```

(i coefficienti ottenuti differiscono dal vettore a ottenuto in precedenza di un arbitrario fattore di scala).

3.10.2 Problemi non lineari

Se le correlazioni fra le variabili non sono lineari, non c'è una procedura generale per risolvere esattamente il problema. Gli algoritmi numerici per "addestrare" un calcolatore a distinguere fra le varie ipotesi sono basati sul concetto di **rete neurale** (vedi, ad esempio, [6]).
Nel software R, il pacchetto "nnet" [9] implementa il più semplice modello di rete neurale, chiamato "perceptron".

Capitolo 4

Modelli di dipendenza

La misura di una grandezza fisica o di un qualunque parametro nelle scienze sperimentali comporta tipicamente un'analisi statistica dei dati sulla base di un modello teorico che includa anche le incertezze nel processo di misura. Spesso il modello implica una dipendenza fra più variabili. Ad esempio, l'altezza h di un grave in caduta libera (nel vuoto) dalla posizione di riposo h_0, dipende dal tempo secondo la legge

$$h = h_0 - g\frac{t^2}{2}$$

Misurando la variabile h per vari valori di t, vorremmo ottenere la miglior stima di g e, al contempo, verificare la compatibilità dei dati col modello ipotizzato.

In questo capitolo tratteremo il problema generale del **fit** dei dati ad un modello di dipendenza, mettendoci nelle seguenti ipotesi:

- si misura una variabile aleatoria y, il cui valore atteso può dipendere da una o più **variabili esplicative**, rappresentate dal vettore $\underline{x}$;

- disponiamo di un modello di dipendenza

$$E(y) = \lambda(\underline{x}; \underline{\theta}) \tag{4.1}$$

  dove $\underline{\theta}$ rappresenta un set di n_p parametri incogniti che vogliamo stimare dai dati stessi;

- il nostro campione è costituito da una serie di N misure $(y_i, (\underline{x})_i)$ dove $N \geq n_p$;

- ci aspettiamo un errore casuale ϵ_i nella misura, indipendente da $\underline{x}$:

$$y_i = \lambda(\underline{x}_i; \underline{\theta}) + \epsilon_i \tag{4.2}$$

  Trascuriamo, per il momento, un eventuale errore sulle variabili esplicative.

4.1 Principio dei minimi quadrati

Gli stimatori di $\underline{\theta}$ possono essere ottenuti dal principio di massima verosimigianza.
Cominciamo col trattare il caso in cui le variabili y_i sono gaussiane e indipendenti; in tal caso la funzione di likelihood è

$$\mathcal{L}(\underline{\theta}) = \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi}\sigma_i} \exp\left(-\frac{(y_i - \lambda(\underline{x_1}; \underline{\theta}))^2}{2\sigma_i^2}\right) \tag{4.3}$$

$$\log\left(\mathcal{L}(\underline{\theta})\right) = k - \sum_{i=1}^{N} \frac{(y_i - \lambda(\underline{x_1}; \underline{\theta}))^2}{2\sigma_i^2} \tag{4.4}$$

Il massimo di $\mathcal{L}$ si ottiene dunque minimizzando il χ^2:

$$(\chi^2)_{min} = \frac{\sum_{i=1}^{N}(y_i - \lambda(\underline{x_i}; \overline{\underline{\theta}}_{MQ}))^2}{\sigma_i^2} \tag{4.5}$$

Nel caso "omoschedastico", ovvero $\sigma_i = \sigma \quad \forall i$, otteniamo gli stimatori minimizzando la somma degli scarti quadratici, giustificando così il **principio dei minimi quadrati**, già postulato da Gauss.

Nel caso $N > n_p$, la soluzione dell'eq. 4.5 ci permette, oltre che di stimare i parametri incogniti, di effettuare un test χ^2 di bontà del fit. Nell'ipotesi nulla (ovvero, assumendo che il nostro modello 4.1 sia corretto) possiamo aspettarci che $(\chi^2)_{min}$ segua la distribuzione di Pearson con $(N - n_p)$ gradi di libertà. Questo è rigorosamente vero nel caso, discusso nel prossimo paragrafo, di un modello lineare, e, anche nel caso non lineare, nel limite asintotico $N \to \infty$.

4.2 Modelli lineari

La soluzione dell'eq. 4.5 consiste in un semplice problema di algebra lineare se il nostro modello ha la forma

$$\lambda(\underline{x}; \underline{\theta}) = \sum_{j=1}^{n_p} a_j(\underline{x})\theta_j \tag{4.6}$$

Si parla in tal caso di "modello lineare". Si noti che la linearità è nei parametri $\underline{\theta}$, non nelle variabili esplicative: anche un polinomio di settimo grado è un modello lineare!
Definendo un vettore $\underline{\lambda}$ tale che $\lambda_i = \lambda(\underline{x}_i)$, possiamo scrivere il χ^2 come

$$\chi^2 = (\underline{y} - \underline{\lambda})^T V_y^{-1} (\underline{y} - \underline{\lambda}) \tag{4.7}$$

dove V_y è la matrice varianza–covarianza delle y_i che nelle nostre ipotesi è una matrice diagonale[1]

[1] la richiesta di indipendenza degli errori può essere rilasciata in quanto l'equazione 4.7 è valida anche nel caso di errori non indipendenti ma descritti da una distribuzione gaussiana multivariata.

Per il modello lineare definiamo una matrice A, detta **matrice del modello**

$$A_{ij} = a_j(\underline{x}_i) \tag{4.8}$$

per cui l'eq. 4.5 diventa

$$(\chi^2)_{min} = (\underline{y} - A\underline{\overline{\theta}}_{MQ})^T V_y^{-1} (\underline{y} - A\underline{\overline{\theta}}_{MQ}) \tag{4.9}$$

la cui soluzione è la trasformazione lineare di $\underline{y}$

$$\underline{\overline{\theta}}_{MQ} = \left(A^T V_y^{-1} A\right)^{-1} A^T V_y^{-1} \underline{y} \ \equiv B\underline{y} \tag{4.10}$$

Gli stimatori ottenuti sono corretti, in quanto

$$E(\underline{\overline{\theta}}_{MQ}) = BE(\underline{y}) = B\underline{\lambda} = BA\underline{\theta} = \left(A^T V_y^{-1} A\right)^{-1} A^T V_y^{-1} A\underline{\theta} = \underline{\theta} \tag{4.11}$$

La matrice varianza–covarianza degli stimatori è data da (cfr eq. 1.56)

$$V_\theta = BV_yB^T = \left(A^T V_y^{-1} A\right)^{-1} \tag{4.12}$$

che in generale non sarà diagonale: gli stimatori dei diversi parametri possono essere correlati. Si noti che

$$\left(A^T V_y^{-1} A\right)_{ij} = \frac{1}{2} \frac{\partial^2 \chi^2}{\partial\theta_i \partial\theta_j}\bigg|_{\underline{\overline{\theta}}_{MQ}} = -\frac{\partial^2 log(\mathcal{L})}{\partial\theta_i \partial\theta_j}\bigg|_{\underline{\overline{\theta}}_{MQ}} \tag{4.13}$$

e dunque V_θ è proprio l'inversa della matrice di informazione di Fisher (eq. 2.31). Gli stimatori così ottenuti sono dunque efficienti.

4.2.1 Il teorema di Gauss-Markov

Nel caso in cui gli errori non siano gaussiani, il principio dei minimi quadrati non coincide più in generale con la soluzione ML. Tuttavia, gli stimatori ottenuti saranno comunque corretti se gli errori ϵ_i hanno media nulla. Inoltre, nell'ipotesi di errori scorrelati e omoschedastici, il teorema di Gauss–Markov [1] dimostra che la soluzione 4.10 è **la più efficiente fra tutti gli stimatori corretti lineari**. Questo giustifica l'uso del metodo dei minimi quadrati anche nel caso non gaussiano. Il $(\chi^2)_{min}$ avrà ancora valore atteso pari a $(N - n_p)$, ma non seguirà in generale la distribuzione di Pearson.

4.2.2 Regressione lineare

Nel caso frequente in cui non conosciamo a priori l'errore σ ma possiamo assumere che sia lo stesso per tutte le misure, si ha $V_y = \sigma^2 I$, e dunque per il calso lineare

$$B = \left(A^T A\right)^{-1} A^T \tag{4.14}$$

La soluzione 4.10 non dipende dunque da σ.
Questo tipo di analisi è chiamata "regressione lineare" ed è molto comune, essendo le ipotesi

fatte, grazie anche al teorema di Gauss–Markov, soddisfatte in molti casi pratici.
Una stima di σ è invece necessaria per stimare l'errore sui parametri:

$$V_\theta = \sigma^2 BB^T = \sigma^2 \left(A^T A\right)^{-1} \tag{4.15}$$

Analogamente allo stimatore 2.10, possiamo usare

$$\overline{\sigma^2} = \frac{\sum_{i=1}^{N}(y_i - \lambda(\underline{x}_1; \overline{\underline{\theta}}_{MQ}))^2}{N - n_p} \tag{4.16}$$

che equivale a fissare χ^2_{min} al suo valore atteso $(N - n_p)$. Si può vedere facilmente che questa stima è corretta e indipendente dalla stima di $\underline{\theta}$.
La variabile

$$t_i = \frac{\left((\overline{\underline{\theta}}_{MQ})_i - \theta_i\right)}{\overline{\sigma(\theta_i)}} \tag{4.17}$$

dove la stima $\overline{\sigma(\theta_i)}$ è ottenuta dalle eq. 4.15 e 4.16, seguirà una distribuzione di Student con $(N - n_p)$ gradi di libertà.

In questo caso non è possibile eseguire un test χ^2 di bontà del fit. Si può comunque fare una diagnostica del modello, testando l'ipotesi che i **residui**

$$r_i = y_i - \lambda(\underline{x}, \overline{\underline{\theta}}_{MQ}) \tag{4.18}$$

siano compatibili, indipendentemennte dal valore di $\underline{x}$, con la distribuzione attesa a media nulla e varianza σ. E' utile quindi visualizzare i residui in funzione delle variabili esplicative. Un test quantitativo per mettere in luce un'eventuale dipendenza residua non tenuta in conto dal modello, consiste in un'analisi della varianza dei residui, divisi in gruppi in base ai valori delle variabili esplicative.

4.2.3 Predizione in base al modello

La soluzione del *fit* può essere utilizzata per estrapolare il modello ad un valore $\underline{x}$' delle variabili esplicative. Per calcolare l'errore sulla nostra predizione, possiamo far uso della formula di propagazione degli errori, che vale esattamente nel caso lineare:

$$\overline{\lambda'} = \sum_{j=1}^{n_p} a_j(\underline{x}')\overline{\theta}_{j\,MQ} \tag{4.19}$$

$$\sigma^2(\overline{\lambda'}) = \sum_{j=1}^{n_p} \left(a_j(\underline{x}')\right)^2 \sigma^2(\theta_j) + \sum_{k \neq l} a_k(\underline{x}')a_l(\underline{x}')cov(\theta_k, \theta_l) \tag{4.20}$$

Nell'ipotesi di errori gaussiani, se la matrice degli errori V_y è nota, anche lo stimatore $\overline{\lambda'}$ seguirà una distribuzione normale.
Nel caso di analisi di regressione lineare, in cui σ è ignota, l'eq. 4.20 è una stima della varianza, e la quantità

$$t(\lambda') = \frac{(\overline{\lambda'} - \lambda(\underline{x}'))}{\overline{\sigma(\overline{\lambda'})}} \tag{4.21}$$

seguirà la statistica di Student con $(N - n_p)$ gradi di libertà. L'intervallo di confidenza, con livello α_{CL}, sulla predizione sarà dunque

$$\overline{\lambda'} \pm \overline{\sigma(\overline{\lambda'})} q_t \left(\frac{(1+\alpha_{CL})}{2}; N - n_p \right) \tag{4.22}$$

Si noti che che $\overline{\lambda'}$ rappresenta lo stimatore del valore atteso di $y(\underline{x})$. L'errore sullo stimatore tende a zero come $1/\sqrt{N}$, tuttavia ogni singola misura di y sarà soggetta ad un errore σ attorno al suo valore atteso.

Nel caso in cui $\underline{x}'$ rappresenti un set di più valori delle variabili esplicative, conviene usare la notazione matriciale $A'_{ij} = a_j(\underline{x}'_i)$:

$$\overline{\underline{\lambda'}} = A'(\underline{x}')\underline{\theta}_{MQ} \tag{4.23}$$

$$V(\lambda') = A'(\underline{x}')V_\theta A'(\underline{x}')^T \tag{4.24}$$

ed otteniamo dunque anche le covarianze fra le predizioni a diversi valori di $\underline{x}$.

4.2.4 Retta dei minimi quadrati

Nel caso del semplice modello

$$\lambda(x; a, b) = a + bx \tag{4.25}$$

la soluzione prende il nome di "retta dei minimi quadrati". Scrivendo esplicitamente le eq. 4.10 e 4.12, si ottengono le formule

$$\overline{a} = \frac{(\sum x_i^2)(\sum y_i) - (\sum x_i)(\sum x_i y_i)}{N \sum x_i^2 - (\sum x_i)^2} \tag{4.26}$$

$$\overline{b} = \frac{N \sum x_i y_i - (\sum x_i)(\sum y_i)}{N \sum x_i^2 - (\sum x_i)^2} \tag{4.27}$$

$$V(a,b) = \sigma_y^2 \begin{pmatrix} \frac{1}{N} + \frac{\overline{x}^2}{\sum(\delta_i)^2} & -\frac{\overline{x}}{\sum(\delta_i)^2} \\ -\frac{\overline{x}}{\sum(\delta_i)^2} & \frac{1}{\sum(\delta_i)^2} \end{pmatrix} \tag{4.28}$$

dove $\overline{x} = (\sum x_i)/N$, $\delta_i = x_i - \overline{x}$. Per l'errore sull'estrapolazione 4.20 si ha

$$\sigma^2(\overline{\lambda'}) = \sigma^2(a + bx') = \overline{\sigma_y^2} \left(\frac{1}{N} + \frac{(x' - \overline{x})^2}{\sum_i (x_i - \overline{x})^2} \right) \tag{4.29}$$

4.2.5 Modelli lineari in R

Nel software *R*, la funzione *lm()* permette di risolvere fits lineari, tramite il calcolo numerico della 4.10. Il modello 4.1 è espresso tramite la sintassi

$$y \sim I(a_1(x)) + I(a_2(x)) + \ldots$$

che di default aggiunge un parametro intercetta θ_0 (per evitarlo, si usi la sintassi $y \sim 0 + I(a_1(x)) + \dots$).
Ad esempio, per il modello $\lambda = \theta_0 + \theta_1 x^2 + \theta_2[\log(x) + 1]$

$$fit = lm(y \sim I(x^2) + I(\log(x) + 1))$$

La funzione fa per default un'analisi di regressione lineare, assumendo il caso omoschedastico con σ ignota. E' possibile specificare gli errori sulle y_i tramite l'argomento *weights*, che rappresenta il vettore $1/\sigma_i^2$.
L'oggetto restituito da *lm()* può essere processato da diverse funzioni:

coef(fit) ritorna un vettore con le stime dei parametri;

resid(fit) il vettore dei residui;

predict(fit) le predizioni λ_i del fit;

deviance(fit) riporta il χ^2 del fit; se l'argomento *weights* non è specificato, i pesi hanno valore 1 e si ottiene la somma dei residui al quadrato;

vcov(fit) la matrice var.–cov. dei parametri. Si noti che questa viene sempre stimata, anche nel caso di *weights* $\neq 1$, rinormalizzando gli errori su y in modo che χ^2 abbia il suo valore atteso.

Esempio 4.2.1 *Caduta di gravi*
La tabella nel file
/afs/math.unifi.it/service/Rdsets/cadutalibera.rdata
mostra i risultati di un esperimento in cui l'altezza h (in m) di un peso in caduta libera nel vuoto è misurata a diversi tempi t (in s) a partire da una posizione di riposo h_0. Vogliamo misurare l'accelerazione gravitazionale g nell'ipotesi di errori gaussiani e costanti sulle misure di h ed errori trascurabili sulle misure di t.

I dati possono essere descritti dal modello lineare

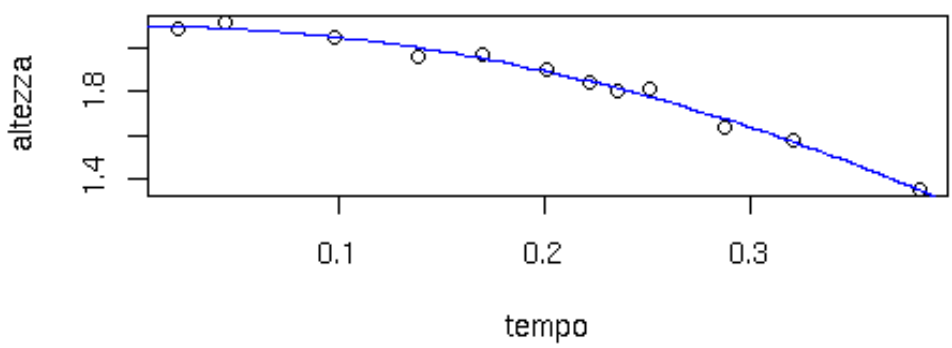
$$h = h_0 - gt^2/2 + \epsilon$$

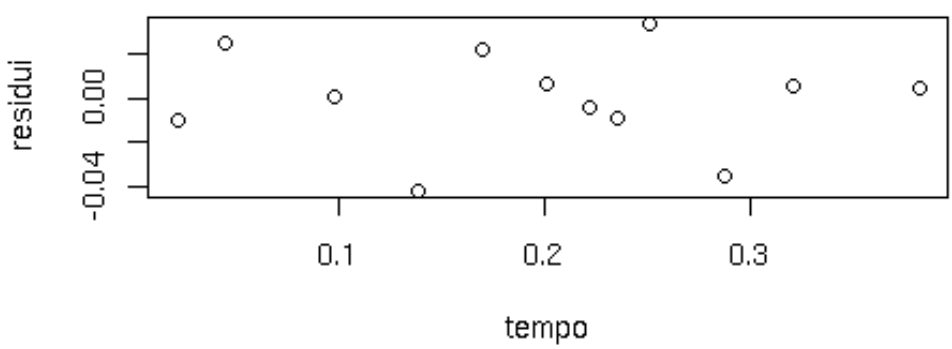
dove ϵ rappresenta l'errore casuale che segue una distribuzione gaussiana con σ ignota. Risolviamo l'analisi di regressione lineare minimizzando lo scarto quadratico medio in funzione dei parametri incogniti h_0 e g:

```
par(mfrow=c(2,1))
cg=read.table("/afs/math.unifi.it/service/Rdsets/cadutalibera.rdata")
attach(cg)
# visualizziamo la dipendenza fra i valori di altezza e tempo
plot(altezza ~ tempo)
n=length(altezza)
# creiamo la matrice del modello
```

```
A= matrix( nrow=n,ncol=2,data=c(rep(1,n),-tempo^2/2))
K = solve(t(A) %*% A)
# ricaviamo il vettore theta con la stima dei parametri h0 e g
theta = K %*% t(A) %*% altezza
# disegniamo la funzione ottenuta dal fit
curve(theta[1]-theta[2]*x^2/2,add=T,col="blue")
# stima dell'errore su h
residui = altezza - A  %*% theta
# visualizziamo i residui in funzione del tempo
plot(residui ~ tempo)
var.h = sum(residui^2)/(n-2)
cat("errore stimato su h:",sqrt(var.h),"\n")
# stima dell'errore sui parametri
V.theta = var.h * K
sigma.theta = sqrt(diag(V.theta))
cat("h0=",theta[1]," +/- ",sigma.theta[1],"\n")
cat("g=",theta[2]," +/- ",sigma.theta[2],"\n")
```

otteniamo
$h_0 = 2.10 \pm 0.01$
$g = 10.2 \pm 0.3$





Il grafico che mostra i residui in funzione di t ci permette di verificare l'assenza di una eventuale dipendenza residua non inclusa nel modello.
La funzione *lm()* di R ci permette di svolgere il calcolo con un'unica chiamata:

```
> summary(lm(altezza ~ I(-tempo^2/2),data=cg))
Call:
lm(formula = altezza ~ I(-tempo^2/2), data = cg)

Residuals:
      Min        1Q    Median        3Q       Max
-0.041615 -0.009098  0.003080  0.010776  0.033711

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    2.09758    0.01057  198.36  < 2e-16 ***
I(-tempo^2/2) 10.24337    0.32325   31.69 2.30e-11 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.02358 on 10 degrees of freedom
Multiple R-Squared: 0.9901,     Adjusted R-squared: 0.9892
F-statistic:  1004 on 1 and 10 DF,  p-value: 2.303e-11
```

Si noti che nella formula che descrive il modello l'intercetta h_0 è implicita; se non avessimo voluto includerla avremmo scritto

```
lm(altezza ~ 0 + I(tempo^2))
```

Test χ^2 sul fit
Eseguiamo un test della bontà del fit sapendo che lo strumento con cui si misurano le altezze ha una risoluzione di 3 cm.
Conoscendo $\sigma_h = 0.03$ m, possiamo ora calcolare il χ^2 del fit e il corrispondente *p-value*:

```
sigma.y = 0.03
chi2 = sum(residui^2)/sigma.y^2
cat("chi2=",chi2," con ",n-2," gradi di liberta'\n" )
cat("p-value = ",1-pchisq(chi2,df=n-2),"\n")
```

Il valore ottenuto *p-value*=0.80 ci conferma la validità del modello.
Gli errori su h possono essere specificati nella funzione *lm()* di R tramite l'argomento *weights* che rappresenta il vettore dei pesi $1/\sigma_h^2$:

```
fit = lm(altezza ~ I(tempo^2/2), weights=rep(1/sigma.y^2,n))
chi2 = deviance(fit)
```

Intervallo di confidenza sulla predizione del fit
Stimiamo ora il valore atteso di h al tempo $t' = 0.6$ s e il corrispondente intervallo di confidenza con livello α=0.95%.

La stima di $E(h(t'))$ è ovviamente

$$\bar{h}(t') = \bar{h_0} - \bar{g}t'^2/2 = A(t')\theta$$

dove $A(t')$ è la matrice del modello calcolata nel punto t' e θ il vettore (h_0, g).
La stima della sua varianza è ottenuta da

$$\bar{\sigma}^2(\bar{h}(t')) = \bar{\sigma}^2(h_0) + (t'^2/2)^2\bar{\sigma}^2(g) + 2(-t'^2/2)\overline{cov}(h_0, g) = A(t')V_\theta A^T(t')$$

(L'espressione matriciale ci permette di estendere le formule al caso in cui t' è un vettore di valori, nel qual caso l'ultima l'espressione diventa la matrice varianza/covarianza delle predizioni $\bar{h}(t')$).
La variabile

$$t(h(t')) = (\bar{h}(t') - E(h(t')))/\bar{\sigma}(\bar{h}(t'))$$

segue una distribuzione di Student con $N - 2$ gradi di libertà. L'intervallo di confidenza è dunque ottenuto come segue:

```
mypred = function(tprime=0.6,alpha=0.95) {
 A.tprime= matrix( nrow=1,ncol=2,data=c( 1,-tprime^2/2))
 h.pred= A.tprime  %*% theta
 sigma.pred = sqrt(A.tprime %*% V.theta  %*% t(A.tprime))
 Nsigma = qt((1+alpha)/2,df=n-2)
 cat("h(",tprime,")=",h.pred, " +/- ",Nsigma*sigma.pred," C.L.=",alpha,"\n")
}
> mypred(0.6,0.95)
h( 0.6 )= 0.2537696  +/-  0.1126419  C.L.= 0.95
```

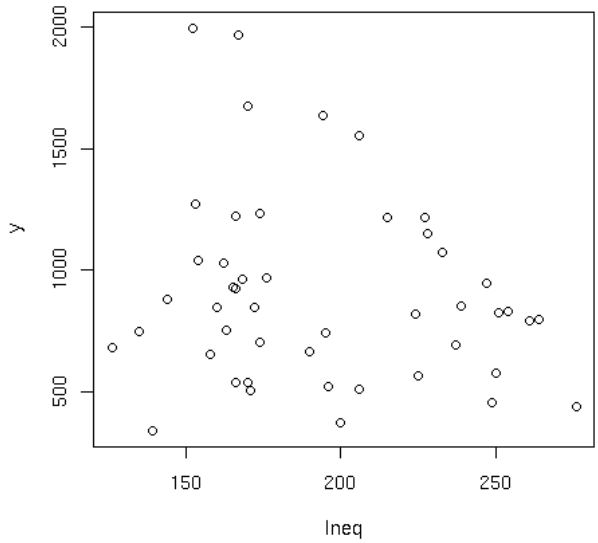
Al solito, R prevede una funzione *predict*() per fare questo calcolo in una sola chiamata:

```
>  myfit = lm(altezza ~ I(-tempo^2/2),data=cg)
>  predict(myfit,newdata=(tempo=0.6),interval="confidence",level=0.95)
          fit       lwr       upr
[1,] 0.2537696 0.1411276 0.3664115
```

L'analisi di regressione lineare è molto usata nelle analisi "esplorative", quando si vogliono studiare empiricamente le possibili dipendenze di una variabile (l'incidenza di una malattia, il tasso di criminalità, il gradimento di un certo prodotto...) da un set arbitrariamente grande di possibili variabili esplicative.

Esempio 4.2.2 *Regressione lineare "esplorativa" nel caso multivariato*
Usando i dati nel dataframe *UScrime* (pacchetto MASS), che contiene una serie di rilevazioni statistiche nei vari stati degli USA, vogliamo determinare quantitativamente la possibile dipendenza del rate di crimini y dal livello di ineguaglianza *Ineq*, tenendo conto o meno della variabile prodotto interno lordo *GDP*. Il grafico

```
attach(MASS)
plot(y ~ Ineq, data=UScrime)
```



non mostra una evidente correlazione fra le due variabili, anche se suggerisce una possibile anticorrelazione. Esploriamo dunque il modello lineare
$y = y_0 + \theta \cdot Ineq + \epsilon$

```
> summary(lm(y ~ Ineq, data=UScrime))
Call:
lm(formula = y ~ Ineq, data = UScrime)
Residuals:
    Min      1Q  Median      3Q     Max
-658.54 -271.38  -30.02  183.75 1017.06

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 1241.773    281.478   4.412 6.33e-05 ***
Ineq          -1.736      1.422  -1.221    0.229
```

Otteniamo effettivamente un valore negativo per θ, tuttavia il test di Student dell'ipotesi $\theta = 0$ dà p–*value* $= 23$ %, e non possiamo dunque affermare di aver messo in evidenza una dipendenza fra le due variabili.
Se ora consideriamo la variabile GDP, notiamo che questa è evidentemente correlata con le altre due variabili:

```
# valutiamo i coefficienti di correlazione lineare
> cor(UScrime[,c("y","Ineq","GDP")])
              y       Ineq        GDP
y     1.0000000 -0.1790237  0.4413199
Ineq -0.1790237  1.0000000 -0.8839973
GDP   0.4413199 -0.8839973  1.0000000
# e il loro errore:
> sqrt( (1-cor(UScrime[,c("y","Ineq","GDP")])^2) / length(UScrime$y))
             y       Ineq        GDP
```

```
y    0.0000000 0.14350851 0.13089192
Ineq 0.1435085 0.00000000 0.06819072
GDP  0.1308919 0.06819072 0.00000000
```

Verifichiamo allora una possibile dipendenza di y da $Ineq$ dopo aver tenuto conto della dipendenza da GDP con una semplice ipotesi lineare:
$y = y_0 + \theta_1 \cdot Ineq + \theta_2 \cdot GDP + \epsilon$

```
> summary(fit <- lm(y ~ Ineq + GDP, data=UScrime))
Call:
lm(formula = y ~ Ineq + GDP, data = UScrime)

Residuals:
    Min      1Q  Median      3Q     Max
-610.09 -193.09  -16.01  108.05  814.69

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -3639.047    968.692  -3.757 0.000503 ***
Ineq            9.364      2.424   3.863 0.000364 ***
GDP             5.192      1.002   5.179 5.32e-06 ***
```

Vista la forte correlazione fra GDP e $Ineq$, ci aspettiamo che i due parametri θ_1 e θ_2 siano pure correlati:

```
> cov2cor(vcov(fit))
            (Intercept)       Ineq        GDP
(Intercept)   1.0000000 -0.9660638 -0.9728075
Ineq         -0.9660638  1.0000000  0.8839973
GDP          -0.9728075  0.8839973  1.0000000
```

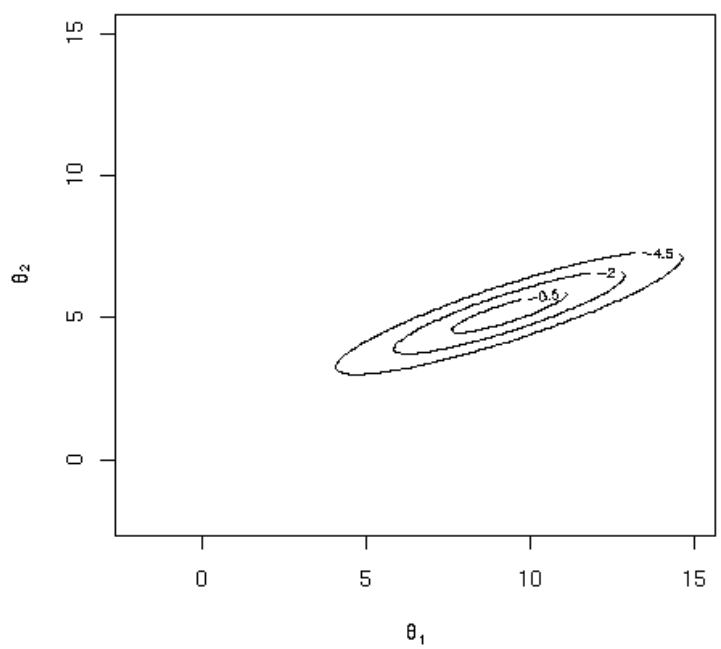
Essendo comunque i due parametri significativamente diversi da 0, possiamo concludere di aver messo in evidenza una correlazione positiva fra y e $Ineq$ per stati con lo stesso prodotto interno lordo. Possiamo visualizzare il risultato disegnando l'intervallo di confidenza per θ_1 e θ_2, che, nell'ipotesi di errori gaussiani, è definito dai punti all'interno dell'ellisse di covarianza per cui

$$\log(L_{max}) - \log(L(\theta_1, \theta_2)) = \frac{1}{2}(\theta - \bar{\theta})^T V_\theta^{-1}(\theta - \bar{\theta}) < N_\sigma^2/2$$

dove L è la funzione di likelyhood e θ è il vettore $(\theta_1\ ,\ \theta_2)$.
Il seguente codice disegna le ellissi corrispondenti a $N_\sigma = 1, 2, 3$, che nel caso bidimensionale corrisponde agli intervalli di confidenza 39.3 %, 86.5 %, 98.9 %

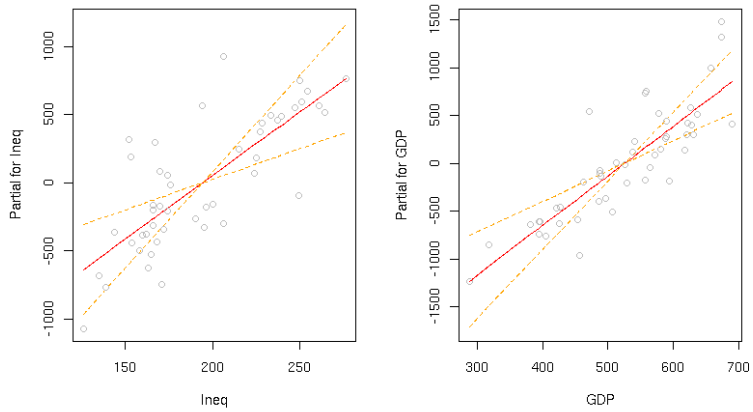
```
mylik = function(t1,t2,theta1,theta2,s1,s2,rho) {
 -1/(2*(1-rho)) * ( (t1-theta1)^2/s1^2 +
                        (t2-theta2)^2/s2^2 -
                    2*rho* (t1-theta1)/s1 *(t2-theta2)/s2)
}
```

```
nsigma=c(1,2,3)
theta1=coef(fit)["Ineq"]
theta2=coef(fit)["GDP"]
s1=sqrt(vcov(fit)["Ineq","Ineq"])
s2= sqrt(vcov(fit)["GDP","GDP"])
rho = cov2cor(vcov(fit))["Ineq","GDP"]
x=seq(-2,15,.1)
y=seq(-2,15,.1)

contour(x,y, outer(x,y,mylik,
  theta1=theta1, theta2=theta2,s1=s1,s2=s2,rho=rho),
  levels = -nsigma^2/2 , xlab=expression(theta[1]),ylab=expression(theta[2]))
```



I residui parziali
$r_{Ineq} = y - (y_0 + \theta_2 \cdot GDP)$
$r_{GDP} = y - (y_0 + \theta_1 \cdot Ineq)$
permettono di visualizzare la dipendenza residua da ogni singola variabile dopo aver tenuto conto delle altre variabili del fit. Possono essere visualizzati tramite la funzione *termplot*()

```
termplot(fit,partial=T,se=T)
```



Sottolineiamo infine il fatto che i risultati di queste analisi di regressione possono mettere in evidenza le mutue dipendenze fra le variabili, ma non potranno mai dimostrare un rapporto di causa–effetto! Il modello deterministico delle dipendenze è lasciato all'interpretazione dell'analista...

4.2.6 Analisi della covarianza

Nelle analisi di tipo esplorativo può aver senso chiedersi se i dati mostrano o meno una dipendenza significativa dalle variabili esplicative. Se abbiamo una variabile esplicativa discreta W, possiamo suddividere i nostri dati in tanti gruppi quanti sono i possibili valori di W ed eseguire un'analisi della varianza (cfr. par. 3.8). Nel caso più generale di un fit lineare con un numero qualsiasi di variabili esplicative, possiamo estendere l'analisi della varianza sostituendo i valor medi di ciascun gruppo con i valori λ_i predetti dal fit

$$\begin{aligned} Q &= \sum_{i=1}^{N}(y_i - \overline{y})^2 = \\ &= \sum_{i=1}^{N}\left[(y_i - \lambda_i) + (\lambda_i - \overline{y})\right]^2 = \\ &= \sum_{i=1}^{N}(y_i - \lambda_i)^2 + \sum_{i=1}^{N}(\lambda_i - \overline{y})^2 + 2\sum_{i=1}^{N}(y_i - \lambda_i)(\lambda_i - \overline{y}) \end{aligned} \tag{4.30}$$

Si dimostra che anche in questo caso il terzo termine si annulla, e dunque

$$Q = Q_E + Q_V \tag{4.31}$$

dove $Q_E = \sum_{i=1}^{N}(y_i - \lambda_i)^2$ quantifica gli scarti rispetto al valore atteso dal fit, e
$Q_V = \sum_{i=1}^{N}(\lambda_i - \overline{y})^2$ le variazioni dovute al modello di dipendenza.
Nell'ipotesi nulla ($E(y)$ indipendente da $\underline{x}$), assumendo errori gaussiani su y, Q_E/σ_y^2 e Q_V/σ_y^2 seguono una distribuzione χ^2 con, rispettivamente, $(N - n_p)$ e $(n_p - 1)$ gradi di libertà.

Questa **analisi della covarianza** (ANCOVA), che possiamo pensare come un limite continuo della ANOVA, consiste dunque in un test di Fisher con con $f_1 = (n_p - 1)$, $f_2 = (N - n_p)$ su

$$F = \frac{Q_V}{(n_p - 1)} \frac{(N - n_p)}{Q_E} \tag{4.32}$$

Per quantificare la dipendenza fra y e $\underline{x}$ viene spesso usato anche il parametro

$$R^2 = \frac{Q_V}{Q} \tag{4.33}$$

che è compreso fra 0 (dipendenza minima) e 1 (dipendenza massima). Si noti che per la retta dei minimi quadrati

$$(R^2)_{rmq} = \frac{\left[\sum_i (x_i - \overline{x})(y_i - \overline{y})\right]^2}{\left(\sum_i (x_i - \overline{x})^2\right)\left(\sum_i (y_i - \overline{y})^2\right)} \tag{4.34}$$

formalmente analogo al quadrato dello stimatore del coefficiente di correlazione.
Tuttavia il solo valore di R^2, la cui statistica dipende dal caso in questione, da solo non basta a dare un'interpretazione probabilistica del risultato come nel caso della ANCOVA, il cui p–value rappresenta la probabilità che, nell'ipotesi nulla, si possa ottenere un F maggiore di quello osservato.
Nel software R il comando summary() sul risultato di lm() riporta una tabella con il risultato dell'analisi della covarianza, come pure il valore di R^2.

Una variante della ANCOVA è usata anche per confrontare, sullo stesso campione di dati, un modello $\lambda_1(\underline{x})$ a n_{p1} parametri con un modello $\lambda_2(\underline{x}) = \lambda 1(\underline{x}) + f_2(\underline{x})$ che aggiunge al precedente alcuni parametri: $n_{p2} > n_{p1}$. Ad esempio, potremmo pensare di aggiungere un termine proporzionale a t^3 nel nostro esempio di caduta dei gravi 4.2.1. La somma dei residui quadratici Q_E sicuramente diminuisce, poiché la minimizzazione ha un grado di libertà supplementare. Possiamo chiederci se questa diminuzione sia significatva o meno, per capire se valga la pena aggiungere un parametro al modello. Anche in questo caso, possiamo dividere Q_E in due termini distinti:

$$\begin{aligned} Q_{E1} &= \sum_{i=1}^{N} (y_i - \lambda_{i1})^2 = \\ &= \sum_{i=1}^{N} \left[(y_i - \lambda_{i2}) + (\lambda_{i2} - \lambda_{i1})\right]^2 = \\ &= Q_{E2} + Q_\delta \end{aligned} \tag{4.35}$$

dove $Q_\delta = \sum_{i=1}^{N} (\lambda_{i2} - \lambda_{i1})^2$. Prendiamo come ipotesi nulla il caso in cui il primo modello è già corretto e il secondo non porta dunque cambiamenti significativi. I valori λ_{i1} sono già stime corrette di $E(y_i)$, come pure λ_{i2}. Ci aspettiamo dunque:
$Q_{E1}/\sigma_y^2 \sim \chi^2(N - n_{p1})$, $Q_{E2}/\sigma_y^2 \sim \chi^2(N - n_{p2})$, e dunque
$Q_\delta/\sigma_y^2 \sim \chi^2(n_{p2} - n_{p1})$. Nell'ipotesi nulla le differenze fra le due soluzioni sono dovute solo alle

fluttuazioni degli errori su y, e non ad un miglioramento del modello di dipendenza. Testiamo allora l'ipotesi nulla con un test di Fisher con $f_1 = (n_{p2} - n_{p1})$, $f_2 = (N - n_{p2})$ su

$$F = \frac{Q_\delta}{(n_{p2} - n_{p1})} \frac{(N - n_{p2})}{Q_{E2}} \tag{4.36}$$

Esempio 4.2.3 *Confronto fra modelli empirici*
Usando i dati nel dataframe *Rubber* (pacchetto MASS), valutiamo l'ipotesi nulla che la variabile *loss* non dipenda dalle variabili *hard* e *tens*.
Si vogliono inoltre confrontare i modelli

$$M1 : loss = \theta_0 + \theta_1 hard + \theta_2 tens + \epsilon$$

$$M5 : loss = \theta_0 \theta_1 hard + \theta_2 tens + \theta_3 tens^2 + \theta_4 tens^3 + \theta_5 tens^4 + \theta_6 tens^5 + \epsilon$$

determinando se il secondo modello migliora significativamente la stima dei valori attesi $loss(hard, tens)$.

Testiamo l'ipotesi nulla tramite un'analisi della covarianza del modello M1.

$$Q = \sum_i (y_i - \bar{y})^2 = \sum_i (y_i - \lambda_i)^2 + \sum_i (\lambda_i - \bar{y})^2 = Q_w + Q_b$$

dove λ_i sono i valori ottenuti dal fit lineare variando gli n_p parametri.
Nell'ipotesi nulla, la variabile

$$F = \frac{Q_b/(n_p - 1)}{Q_w/(N - n_p)}$$

segue una distribuzione di Fisher con $(n_p - 1), (N - n_p)$ gradi di libertà.

```
library(MASS)
fit1=lm(loss~hard + tens,data=Rubber)
np=3
N=length(Rubber$loss)
Qb=sum((fit1$fitted.values-mean(Rubber$loss))^2)
Qw=deviance(fit1)
F = Qb/(np-1) / (  Qw/(N-np) )
cat("F= ",F," p-value=",1-pf(F,df1=np-1,df2=N-np),"\n")
```

Il valore del *p–value*$=1.8 \cdot 10^{-11}$ dimostra la dipendenza fra le variabili, escludendo l'ipotesi nulla.
Si noti che il risultato di questa analisi è riportato anche dalla chiamata $summary(fit1)$.
Si noti anche che nel caso di un singolo parametro di dipendenza (retta dei minimi quadrati),

$$y = \theta_0 + \theta_1 g(x) + \epsilon$$

l'analisi della covarianza è equivalente al test di Student dell'ipotesi $\theta_1 = 0$, con $F = t^2$, ad esempio nel caso

```
> summary(lm(loss~tens,data=Rubber))

Call:
lm(formula = loss ~ tens, data = Rubber)

Residuals:
     Min       1Q   Median       3Q      Max
-155.640  -59.919    2.795   61.221  183.285

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 305.2248    79.9962   3.815 0.000688 ***
tens         -0.7192     0.4347  -1.654 0.109232
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 85.56 on 28 degrees of freedom
Multiple R-Squared: 0.08904,    Adjusted R-squared: 0.0565
F-statistic: 2.737 on 1 and 28 DF,  p-value: 0.1092
```

si ha $t = -1.654, F = 2.73 = t^2$ e il *p–value* di entrambi i tests è pari a 0.1092.

Per il confronto fra i modelli M1 e M5, possiamo usare ancora un'analisi di covarianza scomponendo i residui del primo fit:

$$Q_{w1} = \sum_i (y_i - \lambda_{i1})^2 = \sum_i (y_i - \lambda_{i5})^2 + \sum_i (\lambda_{i1} - \lambda_{i5})^2 = Q_{w5} + Q_b$$

$$F = \frac{Q_b/(n_{p5} - n_{p1})}{Q_{w5}/(N - n_{p5})}$$

```
library(MASS)
N=length(Rubber$loss)
fit1=lm(loss~hard + tens,data=Rubber)
np1=3
fit5=lm(loss~hard + tens + I(tens^2) + I(tens^3)+ I(tens^4)+ I(tens^5),data=Rubber)
np5=7
Qb=sum((resid(fit1)-resid(fit5))^2)
Qw5=deviance(fit5)
F = Qb/(np5-np1) / (  Qw5/(N-np5) )
cat("F= ",F," p-value=",1-pf(F,df1=np5-np1,df2=N-np5),"\n")
```

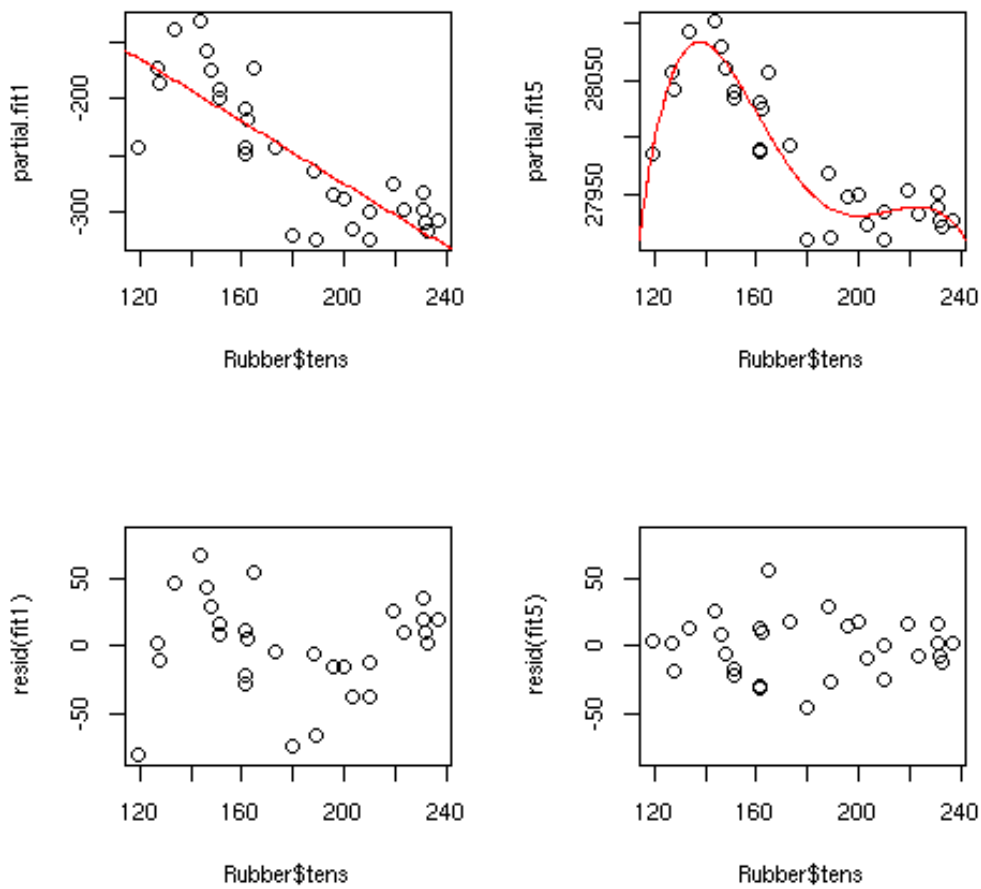
Lo stesso risultato è riprodotto da

```
> anova(fit1,fit5)
Analysis of Variance Table

Model 1: loss ~ hard + tens
```

```
Model 2: loss ~ hard + tens + I(tens^2) + I(tens^3) + I(tens^4) + I(tens^5)
  Res.Df   RSS Df Sum of Sq      F    Pr(>F)
1     27 35950
2     23 12884  4     23065 10.294 6.286e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Il modello M5 è dunque significativamente migliore di M1, come si può anche valutare visivamente dal confronto dei residui:

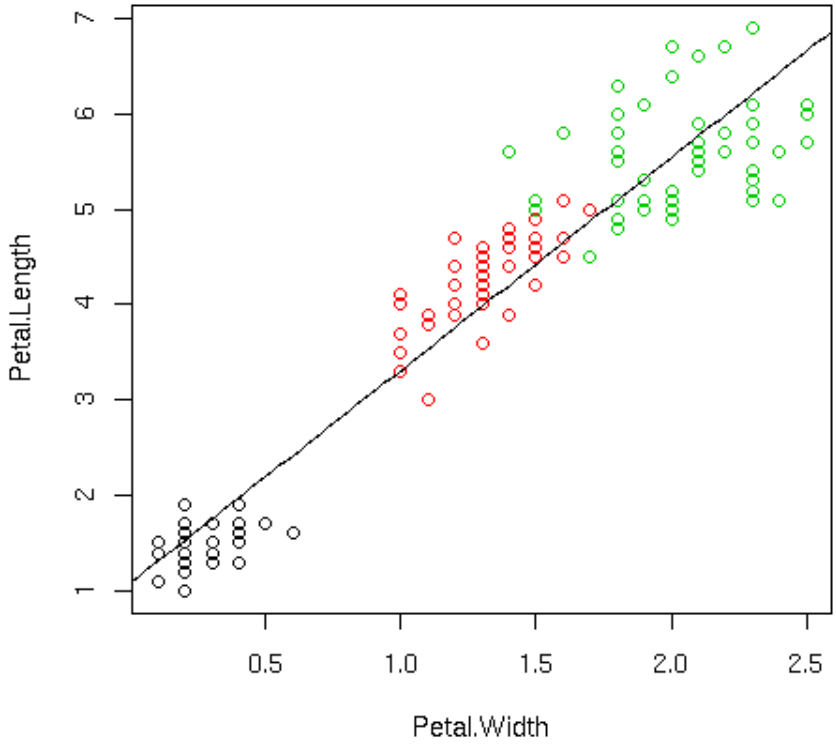
```
par(mfrow=c(2,2))
library(MASS)
N=length(Rubber$loss)
fit1=lm(loss~hard + tens,data=Rubber)
fit5=lm(loss~hard + tens + I(tens^2) + I(tens^3)+ I(tens^4)+ I(tens^5),data=Rubber)
partial.fit1=Rubber$loss - ( coef(fit1)[1] + coef(fit1)[2]*Rubber$hard)
partial.fit5=Rubber$loss - ( coef(fit5)[1] + coef(fit5)[2]*Rubber$hard)
plot( partial.fit1 ~ Rubber$tens)
curve(coef(fit1)[3]*x,add=T,col="red" )
plot( partial.fit5 ~ Rubber$tens)
curve(coef(fit5)[3]*x+coef(fit5)[4]*x^2+coef(fit5)[5]*x^3+
      coef(fit5)[6]*x^4+coef(fit5)[7]*x^5,add=T,col="red" )
plot(resid(fit1) ~ Rubber$tens,ylim=c(-80,80))
plot(resid(fit5) ~ Rubber$tens,ylim=c(-80,80))
par(mfrow=c(1,1))
```



Esempio 4.2.4 *Regressione lineare con variabili categoriali*
Supponiamo di voler modellizare la dipendenza della variabile $Petal.Length$ dalla variabile $Petal.Width$ per i dati nel dataframe iris. I dati suggeriscono una dipendenza lineare

$$Petal.Length = \theta_0 + \theta_1 Petal.Width + \epsilon$$

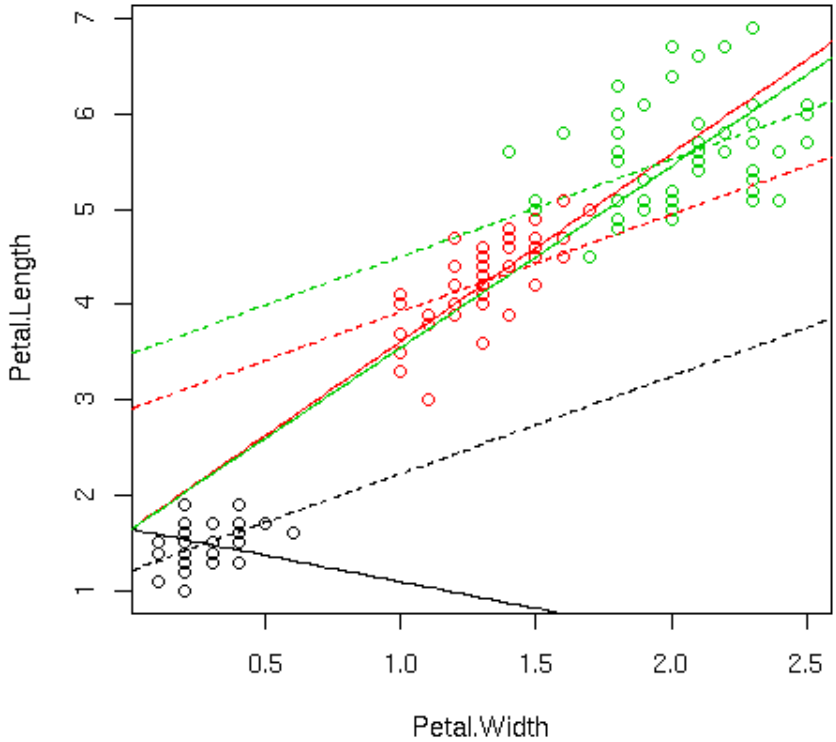
```
plot(Petal.Length ~Petal.Width,data=iris,col=palette()[iris$Species])
lm1=lm(Petal.Length ~ Petal.Width,data=iris)
cf=coef(lm1)
curve(cf[1]+cf[2]*x,add=T)
```



Possiamo ora cercare di migliorare il modello tenendo conto di una eventuale dipendenza dei parametri dalla specie (si noti l'argomento $col = palette()[iris\$Species]$ della funzione *plot* che permette di disegnare i punti con colore diverso a seconda della specie).

Le funzioni di R permettono di eseguire un fit in cui uno o più parametri possono dipendere da una variabile categoriale (factor). Possiamo dunque far dipendere θ_0 o θ_1 dalla specie:

```
plot(Petal.Length ~ Petal.Width,data=iris,col=palette()[iris$Species])
# intercetta comune, pendenza dipendente dalla specie
lm2=lm(Petal.Length ~ 1 + Species:Petal.Width,data=iris)
cf=lm2$coefficients
curve(cf[1]+cf[2]*x,add=T,col=1)
curve(cf[1]+cf[3]*x,add=T,col=2)
curve(cf[1]+cf[4]*x,add=T,col=3)

# pendenza comune, intercetta dipendente dalla specie
lm3=lm(Petal.Length ~ Species - 1 + Petal.Width,data=iris)
cf=lm3$coefficients
curve(cf[1]+cf[4]*x,add=T,col=1,lty=2)
curve(cf[2]+cf[4]*x,add=T,col=2,lty=2)
curve(cf[3]+cf[4]*x,add=T,col=3,lty=2)
```



(si noti la sintassi *Species - 1* per ottenere un'intercetta $(\theta_0)_j (j = 1, 3)$ dipendente dalla specie. Se avessimo omesso il termine *-1*, R avrebbe aggiunto un'intercetta comune e il risultato sarebbe stato equivalente, ma i parametri del fit sarebbero stati $(\theta_0)_1, ((\theta_0)_2 - (\theta_0)_1), ((\theta_0)_3 - (\theta_0)_1)$.)

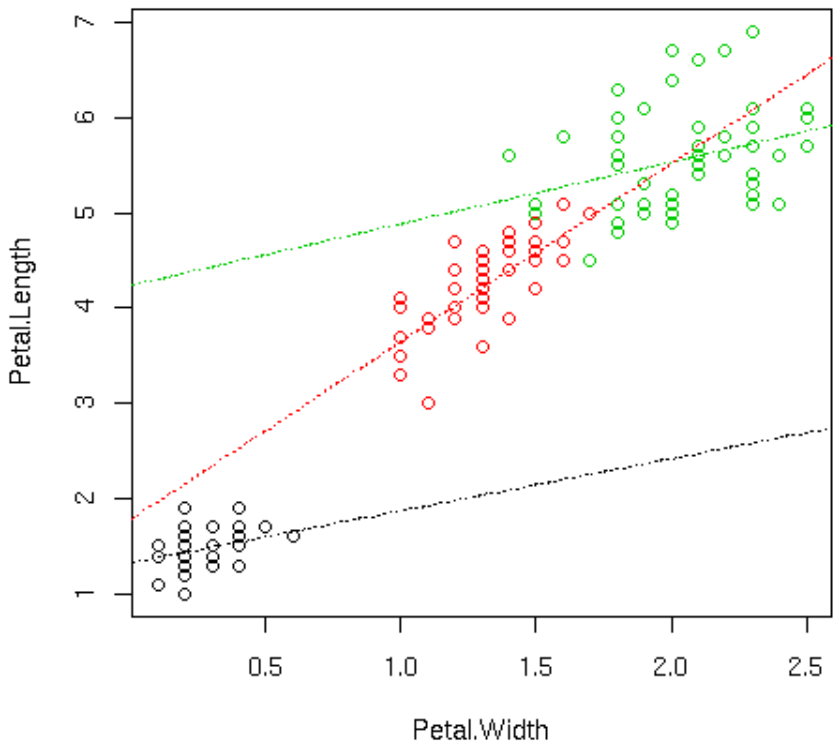
L'analisi della varianza ci suggerisce che per entrambi i modelli $lm2$ e $lm3$, l'aggiunta di 2 parametri rispetto al modello lm1 è giustificata

```
> anova(lm1,lm2)
Analysis of Variance Table

Model 1: Petal.Length ~ Petal.Width
Model 2: Petal.Length ~ 1 + Species:Petal.Width
  Res.Df    RSS  Df Sum of Sq     F    Pr(>F)
1    148 33.845
2    146 25.563   2     8.282 23.65 1.267e-09 ***
---
> anova(lm1,lm3)
Analysis of Variance Table

Model 1: Petal.Length ~ Petal.Width
Model 2: Petal.Length ~ Species - 1 + Petal.Width
  Res.Df    RSS  Df Sum of Sq      F    Pr(>F)
1    148 33.845
2    146 20.833   2    13.011 45.591 4.137e-16 ***
```

e fra i due il modello $lm3$ è evidentemente il migliore, avendo una minore $\sum(y_i - \lambda_i)^2$ a parità di gradi di libertà.

Possiamo infine far dipendere entrambi i parametri dalla specie (equivalente a fare 3 fits indipendenti per ciascuna specie)

```
plot(Petal.Length ~ Petal.Width,data=iris,col=palette()[iris$Species])
# pendenza e intercetta dipendente dalla specie
lm4=lm(Petal.Length ~ Species - 1 + Species:Petal.Width,data=iris)
cf=lm4$coefficients
curve(cf[1]+cf[4]*x,add=T,col=1,lty=3)
curve(cf[2]+cf[5]*x,add=T,col=2,lty=3)
curve(cf[3]+cf[6]*x,add=T,col=3,lty=3)
```



Anche in questo caso, i due parametri aggiuntivi sembrano giustificati:

```
> anova(lm3,lm4)
Analysis of Variance Table

Model 1: Petal.Length ~ Species - 1 + Petal.Width
Model 2: Petal.Length ~ Species - 1 + Species:Petal.Width
  Res.Df     RSS  Df Sum of Sq      F    Pr(>F)
1    146 20.8334
2    144 18.8156   2    2.0178 7.7213 0.0006525 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

4.2.7 Fit di un istogramma

Supponiamo di avere un campione costituito da N misure indipendenti di una variabile aleatoria x, da cui vogliamo stimare i parametri liberi della sua funzione densità $d(x;\underline{\theta})$. La soluzione generale dettata dal principio di massima verosimiglianza consiste nel massimizzare $\mathcal{L} = \prod_{i=1}^{N} d(x_i;\underline{\theta})$ rispetto alle variabili $\underline{\theta}$, cosa che può essere tecnicamnete laboriosa, soprattutto nel caso di molti parametri.

Per semplificare il calcolo, possiamo fare un fit "binnato", ovvero mettendo i dati in un istogramma ed eseguendo un *fit* sui valori dei conteggi k_i in ciascun intervallo I_i. Il modello prevede che ciascun k_i sia distribuito secondo la statistica di Poisson con valore atteso e varianza pari a $\lambda_i(\underline{\theta}) = N \int_{I_i} d(x;\underline{\theta})dx$.

In questo caso la varianza dei valori dipende anch'essa dai parametri, tuttavia nel limite di k grandi, la stima $\overline{\sigma(k_i)} = \sqrt{k_i}$ può essere sufficientemente precisa, e inoltre la distribuzione di k_i tende a quella normale. In questa ipotesi siamo dunque nella stessa situazione del paragrafo 4.1, e possiamo stimare i parametri minimizzando

$$\chi^2 = \frac{\sum_{i=1}^{N} \left(k_i - (N \int_{I_i} d(x;\underline{\theta})dx)\right)^2}{k_i} \tag{4.37}$$

Nel caso di un modello lineare, possiamo dunque risolvere il problema analiticamente utilizzando la semplice soluzione 4.10.

Il numero di gradi di libertà del *fit* sarà $(N - n_p - 1)$, avendo usato i dati sia per stimare gli n_p parametri che per normalizzare le predizioni al valore N di misure.

Il risultato può essere distorto a causa dell'approssimazione $\sigma(k_i) = \sqrt{k_i}$: il peso $1/\overline{\sigma(k_i)}^2$ sarà sovrastimato per i conteggi che hanno fluttuato verso il basso e viceversa. Tuttavia, il valore $\lambda(x_i;\overline{\underline{\theta}}_{MQ})$ predetto dal fit fornisce una stima più accurata del valore atteso $\lambda(x_i,\underline{\theta})$, poiché si basa su tutti i valori, e sul modello ipotizzato, anziché su un singolo conteggio. Per verificare la possibile distorsione possiamo allora migliorare la stima delle σ_i e ripetere il fit, procedendo in modo iterativo fino alla convergenza di χ^2_{min}.

Esempio 4.2.5 *Fit di un picco gaussiano in presenza di fondo*
Eseguiamo un fit delle misure di energia (in GeV) contenute nel file
/afs/math.unifi.it/service/Rdsets/gaussconfondo.rdata
secondo il modello

$$N = N_b + \alpha \exp(-(E-\mu)^2/(2\sigma^2))$$

dove N è il numero atteso di eventi in un intervallo di energia centrato in E di larghezza 0.2 GeV. Vogliamo ricavare la migliore stima di N_b e α sapendo che $\mu = 5.2$ GeV, $\sigma = 0.3$ GeV, ed eseguire un test di bontà del fit. Notiamo che il modello è lineare nei due parametri incogniti.

In questo caso ci aspettiamo che i conteggi k_i misurati in ogni intervallo i seguano una distribuzione di Poisson con valore atteso $N(E_i)$. Possiamo dunque dare una prima stima della varianza $\sigma^2(k_i) = k_i$ e ottenere un primo fit minimizzando

$$\chi^2_{(0)} = \frac{\sum_i (k_i - N(E_i;N_b,\alpha))^2}{k_i}$$

Assumendo la correttezza del modello, i valori dei conteggi ottenuti dal fit saranno una miglior stima di $\sigma^2(k_i)$. Procediamo dunque in modo iterativo, fino ad ottenere una convergenza del valore di χ^2:

$$\chi^2_{(r)} = \frac{\sum_i (k_i - N(E_i; N_b, \alpha))^2}{N(E_i; \bar{N}_{b(r-1)}, \bar{\alpha}_{(r-1)})}$$

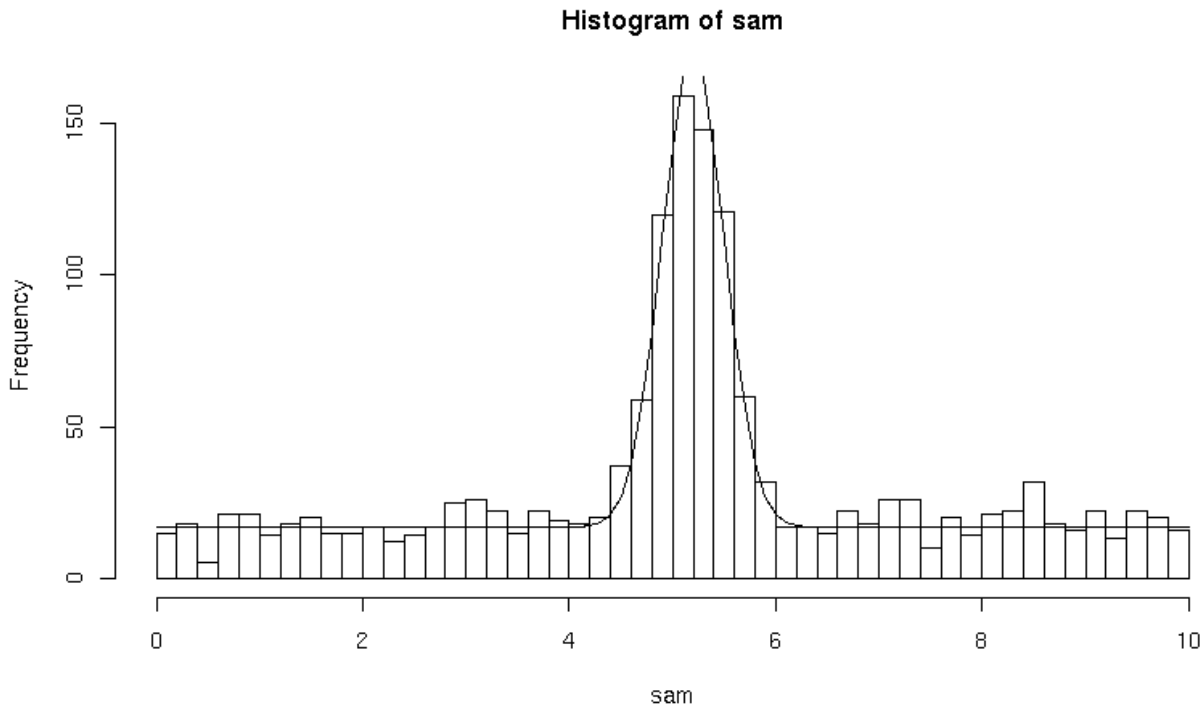
```
histofit = function() {
 sam=scan("/afs/math.unifi.it/service/Rdsets/gaussconfondo.rdata")
 iter=0
 mu=5.2
 sigma=0.3
 h=hist(sam,breaks=seq(0,10,0.2))
 x=h$mids
 k=h$counts
 n=length(k)
 deltax=h$breaks[2]-h$breaks[1]

 chisq=99999
 chisqold=2*chisq
 deltachi=length(scan)/50
 while ((chisqold-chisq)>deltachi ) {
  chisqold=chisq

  cat("######### Iterazione ",iter," #########\n")
  if (iter  == 0) {
   dk2=k
  }
  else {
   dk2=predict(fit)
  }
  fit = lm( k ~ I(dnorm(x, mean=mu, sd=sigma)*deltax), weights=1/dk2 )

  theta = coef(fit)
  cat("theta=",theta,"\n")
  cat(" +/- ",sqrt(diag(vcov(fit))),"\n")

  curve( theta[1] + theta[2] *dnorm(x, mean=mu, sd=sigma)*deltax,
         0,10,add=T,col=(iter+1) )

  chisq = deviance(fit)
  cat ("chi2=",chisq," con ",n-2," g.d.l. p-value=",1-pchisq(chisq,df=n-2),"\n")
  iter=iter+1
  readline()
 }
}
> histofit()
######### Iterazione  0  #########
```

```
theta= 16.91985 588.4278
 +/-  0.794874 34.10682
chi2= 77.57949  con  48  g.d.l. p-value= 0.004377989

######### Iterazione  1  #########
theta= 18.59739 582.1304
 +/-  0.7231749 31.07135
chi2= 63.74022  con  48  g.d.l. p-value= 0.06366115

######### Iterazione  2  #########
theta= 18.60706 581.6469
 +/-  0.7249509 29.84796
chi2= 58.38637  con  48  g.d.l. p-value= 0.1448092

######### Iterazione  3  #########
theta= 18.60719 581.6407
 +/-  0.7249744 29.83274
chi2= 58.36121  con  48  g.d.l. p-value= 0.1453228
```

Si noti la differenza di *p–value* fra il primo fit, in base al quale avremmo escluso la validità del modello, e l'ultimo valore assai più ragionevole.
Notiamo anche che per la nostra predizione abbiamo approssimato l'integrale della funzione densità attesa in ciascun intervallo con il valore calcolato al centro dell'intervallo. Possiamo, pena un maggior dispendio di CPU, evitare questa approssimazione, calcolando numericamente l'integrale tramite la funzione *integrate()* di R:

```
x2=c()
for (i in 1:n) {
 x2[i]= integrate (dnorm , h$breaks[i], h$breaks[i+1], mean=mu, sd=sigma )$value
}
fit = lm( k ~ x2, weights=1/dk2 )
```

La funzione così modificata trova effettivamente un miglior $\chi^2_{min} = 57.2$, corrispondente a *p–value*=17%



4.2.8 Errori sulle variabili esplicative

Finora abbiamo trascurato un'eventuale incertezza sperimentale su una variabile esplicativa x. Ma queste possono essere trascurate rispetto alle incertezze su y solo se

$$\left|\frac{d\lambda(x)}{dx}(x_i)\right|^2 \sigma^2(x_i) \ll \sigma^2(y_i) \tag{4.38}$$

In caso contrario, la soluzione esatta consiste nel massimizzare la funzione di likelihood che tenga conto anche degli errori su x. Tuttavia, se gli errori su x e y sono indipendenti, una ragionevole soluzione approssimata consiste nel risolvere il problema del minimo χ^2 4.5 sostituendo $\sigma^2(y_i)$ con

$$\sigma'^2(y_i) = \sigma^2(y_i) + \left|\frac{d\lambda(x)}{dx}(x_i)\right|^2 \sigma^2(x_i) \tag{4.39}$$

propagando dunque a y l'errore su x nell'approssimazione lineare.

Esempio 4.2.6 *Analisi dello spazio di frenata in funzione della velocità*
I dati nella tabella del file
/afs/math.unifi.it/service/Rdsets/frenata.rdata
rappresentano valori, misurati in metri con una risoluzione di 0.7 m, dello spazio f necessario alla frenata di un' automobile lanciata a velocità v. La velocità è ricavata dalla misura del tempo t (misurato con risoluzione 0.01 s) necessario ad attraversare una distanza di 10 m. Vogliamo testare il modello

$$f = Pv^2$$

dando la migliore stima del parametro P.

Cominciamo col visualizzare i dati in un grafico, mostrando anche gli errori sui valori tramite rettangoli di larghezza $\pm\sigma$ attorno ai valori misurati:

```
fr = read.table("/afs/math.unifi.it/service/Rdsets/frenata.rdata")
sigma.f =  0.7
dt = 0.01
t = 10 / fr$v
dv = fr$v  * dt / t
n=nrow(fr)
plot( frenata ~ v, xlim=c(0,70),data=fr)
deltaf = rep(sigma.f , n)
symbols(fr$v, fr$frenata,fg="blue",
        rectangles=matrix(ncol=2,c(2*dv,2*deltaf)),inches=FALSE,add=T)
```

Notiamo come gli errori su v siano trascurabili rispetto ai quelli su f per bassi valori di v, e siano invece dominanti per alti valori di v. Non possiamo dunque trascurare uno dei due errori (facendo eventualmente un fit di v in funzione di f). Usiamo dunque la procedura 4.39, stimando la derivata da un primo fit in cui si trascurano gli errori su v:

```
fit1 = lm( frenata ~ 0 + I(v^2), data=fr, weights = 1/deltaf^2)
chisq = deviance(fit1)
cat ("primo fit: chi2=",chisq," con ",n-1," g.d.l. p-value=",
       1-pchisq(chisq,df=n-1),"\n")
P = coef(fit1)[1]
curve( P * x^2, col="red", add=T)
```

e teniamo poi conto dell'errore su v:

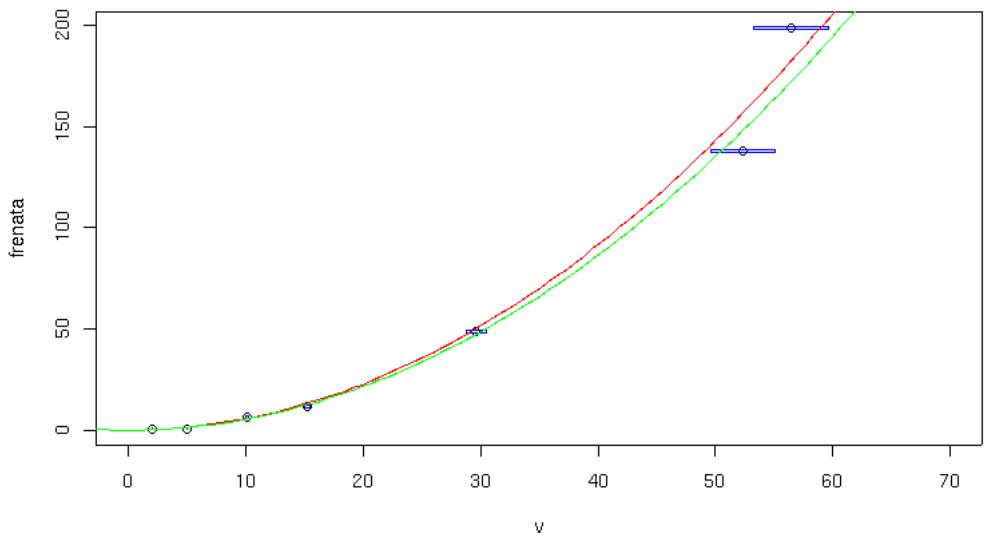
$$\sigma'^2(f_i) = \sigma^2(f_i) + (2Pv)^2\sigma^2(v_i)$$

```
deltafp2 = deltaf^2 + (2*fr$v*P*dv)^2
fit2 = lm( frenata ~ 0 + I(v^2), data=fr, weights = 1/deltafp2)
P = coef(fit2)[1]
curve( P * x^2, col="green", add=T)
chisq = deviance(fit2)
cat ("secondo fit: chi2=",chisq," con ",n-1," g.d.l. p-value=",
      1-pchisq(chisq,df=n-1),"\n")
```

Possiamo notare l'effetto del calcolo più corretto dei pesi sulla stima e, soprattutto, sull'esito del test χ^2:

$$fit1: \quad P = 0.0572 \pm 0.0002 \qquad \chi^2 = 1276 \text{ con 6 g.d.l., } p\text{–}value = 0\%$$

$$fit2: \quad P = 0.054 \pm 0.002 \qquad \chi^2 = 6.62 \text{ con 6 g.d.l., } p\text{–}value = 35.7\%$$



4.3 Modelli non lineari: minimizzazione numerica

Se il modello di dipendenza non è lineare ma gli errori sono gaussiani, gli stimatori ML saranno ancora quelli che minimizzano il χ^2 (eq. 4.5). In generale non c'è una soluzione algebrica, ma possiamo sempre trovare il χ^2_{min} nello spazio n_p–dimensionale dei parametri tramite un algoritmo numerico, facendo ricorso alla matematica dei problemi di **ottimizzazione**[2]
Assumendo che la funzione f da minimizzare sia convessa, e dunque esista un solo minimo, uno dei più semplici algoritmi per trovarlo numericamente è quello di "discesa del gradiente": a partire da un valore iniziale dei parametri $\underline{\theta}_0$, si procede iterativamente nella direzione opposta al gradiente della funzione:

$$\underline{\theta}_{k+1} = \underline{\theta}_k - \gamma \nabla f(\underline{\theta}_k) \tag{4.40}$$

con il criterio di convergenza $|f(\underline{\theta}_{k+1}) - f(\underline{\theta}_k)|/f(\underline{\theta}_k) < \epsilon$
dove γ e ϵ sono costanti prefissate, tipicamente $\ll 1$.
L'idea può essere miglorata utilizzando uno sviluppo al secondo ordine, nel caso unidimensionale:

$$f(\theta + \Delta\theta) \simeq f(\theta) + f'(\theta)\Delta\theta + f''(\theta)\frac{(\Delta\theta)^2}{2} \tag{4.41}$$

$$lim_{\Delta\theta \to 0} \frac{(f(\theta + \Delta\theta) - f(\theta))}{\Delta\theta} = 0 \quad \Longrightarrow \quad \Delta\theta \to -\frac{f'(\theta)}{f''(\theta)} \tag{4.42}$$

da cui il noto **algoritmo di Newton** per trovare gli zeri della derivata

$$\theta_{k+1} = \theta_k - \gamma \frac{f'(\theta)}{f''(\theta)} \tag{4.43}$$

[2]queste problematiche, qui solo accennate, sono trattate nel corso di "Metodi Numerici per l'Ottimizzazione".

Nel caso multidimensionale l'algoritmo diventa

$$\underline{\theta}_{k+1} = \underline{\theta}_k - \gamma[H_f(\underline{\theta}_k)]^{-1}\nabla f(\underline{\theta}_k) \tag{4.44}$$

dove H_f è l'Hessiano della funzione f.

Nella pratica, il calcolo dell'Hessiano risulta la parte più dispendiosa nell'esecuzione dell'algoritmo al computer. Una variante per migliorare l'efficienza dell'algoritmo, noto come **metodo di Gauss–Newton**, consiste nel considerare che la funzione χ^2 da minimizzare è la somma dei quadrati dei residui $r_i(\theta) = (y_i - \lambda_i)/\sigma_i$:

$$f(\theta) = \sum_i r_i^2(\theta) \tag{4.45}$$

$$f'(\theta) = 2\sum_i r_i(\theta) r_i'(\theta) \tag{4.46}$$

$$f''(\theta) = 2\sum_i \left[(r_i'(\theta))^2 + r_i(\theta) r_i''(\theta)\right] \tag{4.47}$$

a approssimando al primo ordine la $r_i(\theta)$, otteniamo

$$\frac{f'(\theta)}{f''(\theta)} \simeq \frac{\sum_i r_i(\theta) r_i'(\theta)}{\sum_i (r_i'(\theta))^2} \tag{4.48}$$

che ci permette di calcolare la 4.43 senza dover stimare le derivate seconde. La formula nel caso multivariato diventa

$$[H_f(\underline{\theta}_k)]^{-1}\nabla f(\underline{\theta}_k) \simeq \left[J(\underline{\theta})^T J(\underline{\theta})\right]^{-1} J(\underline{\theta})^T \underline{r}(\underline{\theta}) \tag{4.49}$$

dove $\underline{r}$ è il vettore con componenti $r_i(\underline{\theta})$ e J è lo Jacobiano $(J)_{ij} = \partial r_i/\partial\theta_j$.

Un'altra famiglia di algoritmi derivati da quello di Newton sono i cosiddetti "quasi–Newton", in cui le variazioni dell'Hessiano ad ogni iterazione vengono stimate con formule approssimate. Il più noto di questi algoritmi è chiamato BFGS dal nome dei suoi autori[3].

Il metodo di Newton e i suoi derivati risultano molto indicati nel caso di stimatori ML nel limite di alta statistica, poiché in tal caso dalla eq. 2.36 ci si aspetta

$$lim_{N\to\infty}\ \chi^2(\theta) = \chi^2_{min} + \frac{(\theta - \overline{\theta}_{ML})^2}{\sigma^2_{\overline{\theta}}} \tag{4.50}$$

Nel limite, la stima della derivata seconda (o dell'Hessiano nel caso multivariato) del χ^2 ci permette di stimare l'errore sullo stimatore:

$$\sigma^2_{\overline{\theta}} \simeq 2/(\frac{\partial^2\chi^2}{\partial\theta^2}) \tag{4.51}$$

$$V_{\overline{\theta}} \simeq 2(H_{\chi^2})^{-1} \tag{4.52}$$

[3]Broyden[2], Fletcher, Goldfarb e Shanno

Ricordiamo dal par. 2.8.2 che nel caso di bassa statistica l'intervallo di confidenza corrispondente a $\pm N_\sigma$ deviazioni standard normali può essere stimato dai valori per cui
$\log \mathcal{L}(\theta) > \log \mathcal{L}(\overline{\theta}) - N_\sigma^2/2$, ovvero
$\chi^2(\theta) < \chi^2_{min} + N_\sigma^2$
Nel software R, la funzione nls() permette di risolvere fits con un generico modello non lineare. L'algoritmo di default è quello di Gauss–Newton, ma è possibile specificare un metodo alternativo tramite l'argomento "algorithm". Gli errori sugli stimatori sono ottenuti dall'inverso dell'Hessiano.
La funzione optim() implementa diversi algoritmi di ottimizzazione, fra cui il BFGS.

Nell'uso pratico di questi algoritmi bisogna ricordare che la procedura può fallire in caso di più minimi locali o di discontinuità della funzione. Per evitare false soluzioni, è necessario scegliere con cura i valori di partenza dei parametri e i limiti entro cui effettuare la ricerca del minimo, e verificare visivamente il risultato facendo il grafico della funzione χ^2 intorno al minimo ottenuto.
Si consiglia inoltre di scegliere le unità dei parametri in modo che abbiano un ordine di grandezza paragonabile, al fine di evitare che gli errori di approssimazione nel calcolo numerico dell'inverso dello Jacobiano o dell'Hessiano possano far fallire l'algoritmo di ottimizzazione.

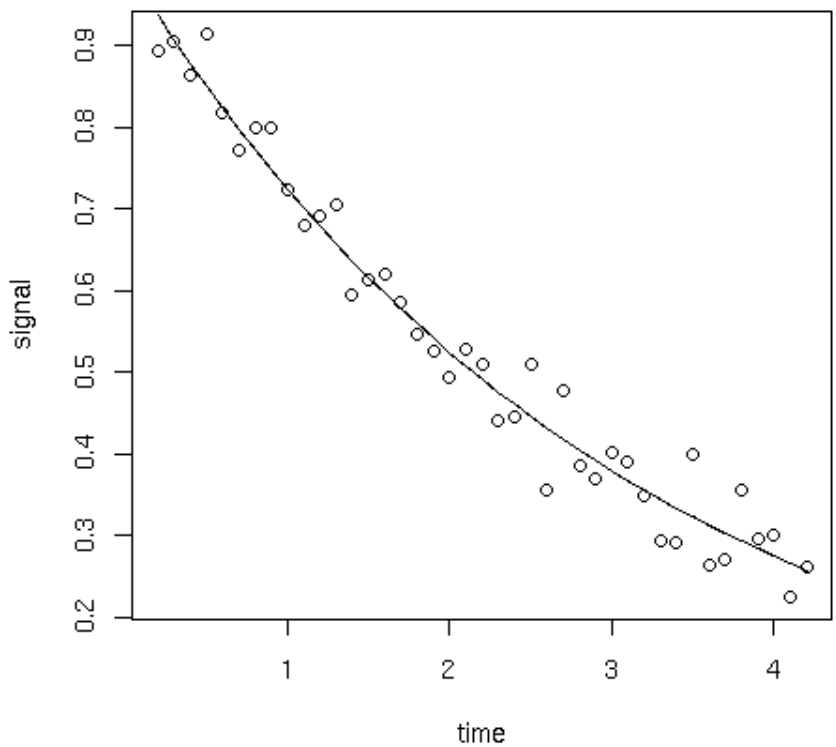
Esempio 4.3.1 *Intensità di una sorgente radioattiva*
I valori nel file
/afs/math.unifi.it/service/Rdsets/decay.rdata
rappresentano la variazione di intensità nel tempo di una sorgente radioattiva, che ci aspettiamo seguire la legge

$$I = I_0 e^{-t/\tau}$$

Sapendo che $I_0 = 1$ (le misure sono relative al tempo 0) e assumendo errori gaussiani sulle misure di I, ricaviamo il parametro τ e l'intervallo di confidenza con CL α=90% per τ assumendo che sia valida l'approssimazione gaussiana. Valutaremo infine la bontà di questa approssimazione.

Utilizziamo la funzione *nls* di R per questa analisi di regressione non lineare. L'argomento *start* deve contenere un vettore con i parametri liberi e il loro valore di partenza. Ogni parametro deve avere il nome che viene usato nella formula del modello. Nel nostro caso abbiamo un solo parametro che chiamiamo "tau":

```
mydata=read.table('/afs/math.unifi.it/service/Rdsets/decay.rdata')
plot(signal ~ time,data=mydata)
fit=nls(signal ~ exp(-time/tau), data=mydata,
start=c(tau=2) )
lines(mydata$time,predict(fit))
```



Dalla minimizzazione numerica del χ^2 il fit ottiene
$\bar{\tau} = 3.10 \pm 0.06$
La variabile

$$t = (\bar{\tau} - \tau)/\bar{\sigma}_\tau$$

segue una distribuzione di Student con $N-1$ gradi di libertà, l'intervallo di confidenza è quindi

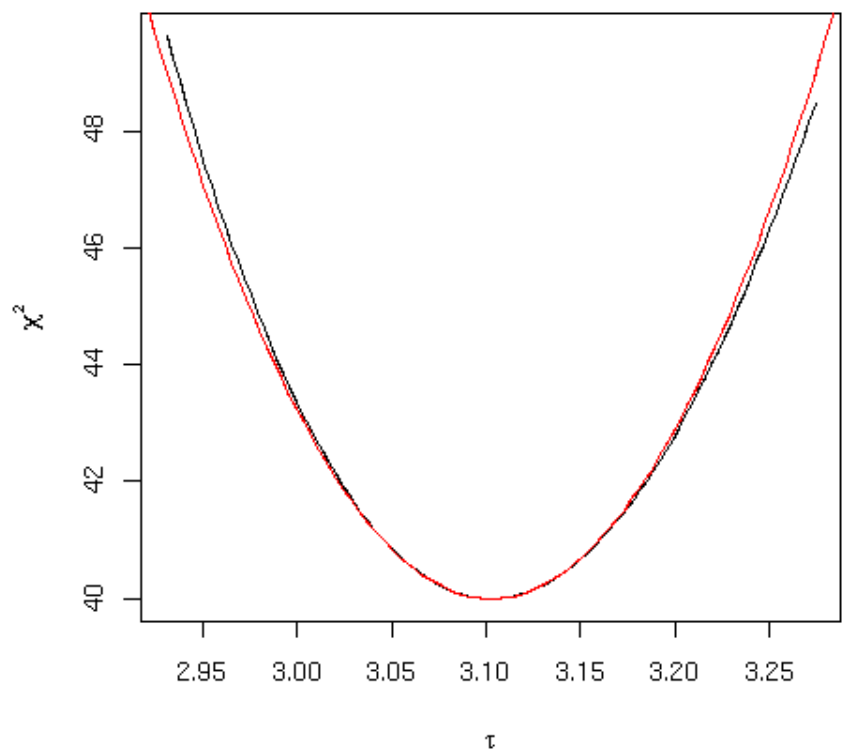
```
N=length(mydata$signal)
tau=fit$m$getPars()[1]
alpha=0.9
dtau=sqrt(vcov(fit)[1,1])
Nsigma = qt((1+alpha)/2,df=N-1)
cat ("tau=",tau," +/- ",Nsigma*dtau,"\n")
```

da cui
$\tau = 3.103 \pm 0.096$ al 90 % C.L.

L'intervallo di confidenza così ottenuto presuppone l'aver assunto una distribuzione di probabilità gaussiana per lo stimatore $\bar{\tau}$, cosa che nel caso non lineare è rigorosamente vera solo nel limite asintotico $N \to \infty$.
Per verificare questa approssimazione, confrontiamo il profilo del χ^2 in funzione di τ, con la predizione gaussiana

$$\chi^2(\tau) - \chi^2(\bar{\tau}) = 2(\log(L(\bar{\tau})) - \log(L(\tau))) = \frac{(\tau - \bar{\tau})^2}{\bar{\sigma}_\tau^2}$$

```
sigmay2=deviance(fit)/(N-1)

chi2.taufit = function(tau) {
 out=c()
 for (i in 1:length(tau) ) {
   out[i]=sum( ( mydata$signal - exp(-mydata$time/tau[i])   )^2  )/sigmay2
 }
 out
}

# profilo effettivo in nero
curve(chi2.taufit(x),tau-3*dtau,tau+3*dtau,
  xlab=expression(tau),ylab=expression(chi^2))
# predizione approssimato in rosso
curve((N-1)+(x-tau)^2/dtau^2,add=T,col="red")
```



Il plot dimostra la validità dell'approssimazione. Se infatti volessimo migliorare la stima dell'intervallo di confidenza, calcolando i valori di τ per cui $\chi^2 = min(\chi^2) + N^2_\sigma(\alpha)$, otterremmo $\tau = 3.103 - 0.094 + 0.098$ al 90 % C.L.
con una variazione della stima degli errori di circa il 2 %.

Caso con due parametri
Ripetiamo ora l'esercizio precedente supponendo di non conoscere il valore di I_0.

Basterà ripetere il procedimento con due parametri liberi

```
mydata=read.table('/afs/math.unifi.it/service/Rdsets/decay.rdata')
plot(signal ~ time,data=mydata)
fit=nls(signal ~ I0*exp(-time/tau), data=mydata,
start=c(I0=1,tau=2) )
print(summary(fit))
lines(mydata$time,predict(fit))
```

da cui si ottiene

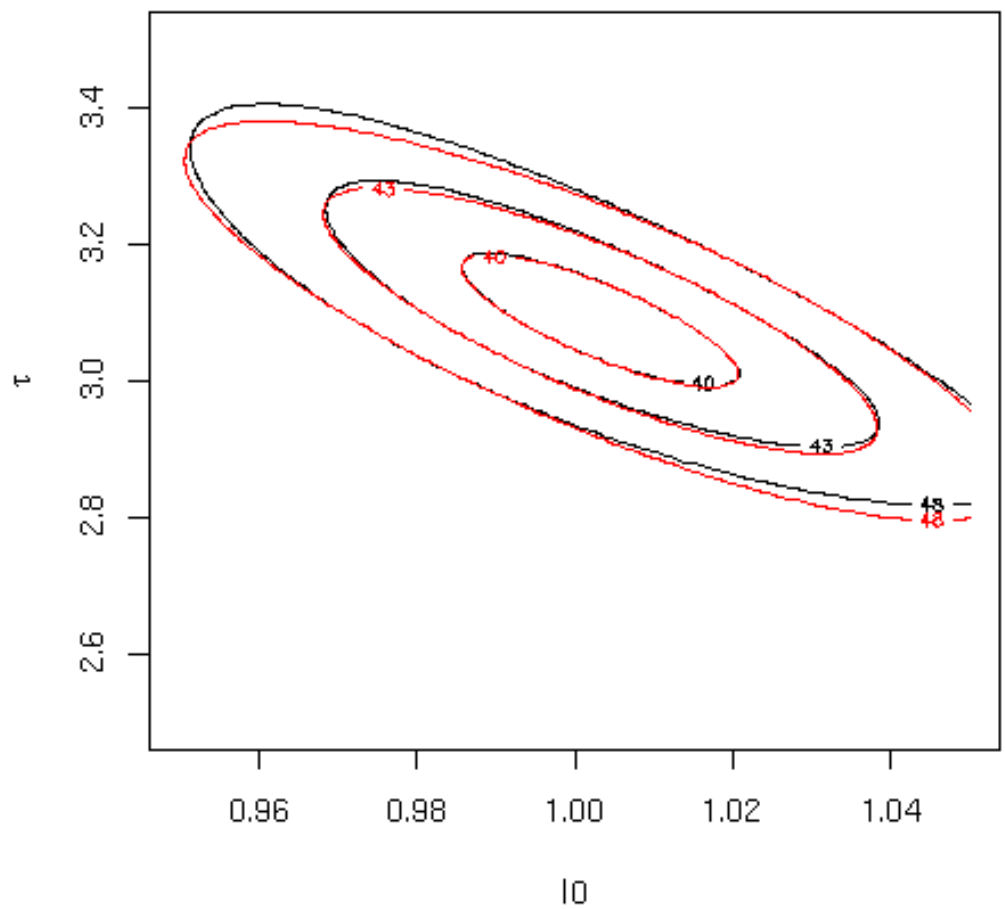
```
Parameters:
    Estimate Std. Error t value Pr(>|t|)
I0   1.00321    0.01757   57.09   <2e-16 ***
tau  3.08820    0.09752   31.67   <2e-16 ***
```

In questo caso, l'approssimazione gaussiana prevede che il profilo del χ^2 sia dato dal logaritmo di una gaussiana bivariata

$$\chi^2(\tau, I_0) - \chi^2(\bar{\tau}, \overline{I_0}) = (\theta - \bar{\theta})^T V_\theta^{-1} (\theta - \bar{\theta}) = \frac{1}{(1-\rho)} \left[\frac{(\tau - \bar{\tau})^2}{\sigma_\tau^2} + \frac{(I_0 - \overline{I_0})^2}{\sigma_{I_0}^2} - \frac{2\rho(\tau - \bar{\tau})(I_0 - \overline{I_0})}{\sigma_\tau \sigma_{I_0}} \right]$$

```
N=length(mydata$signal)
theta=fit$m$getPars()
V.theta=vcov(fit)
dtau=sqrt(diag(vcov(fit)))
sigmay2=deviance(fit)/(N-2)

chi2.taufit2 = function(I0,tau) {
 out=matrix(nrow=length(I0),ncol=length(tau))
 for (i in 1:length(I0) ) {
  for (j in 1:length(tau) ) {
  out[i,j]=sum( ( mydata$signal - I0[i]*exp(-mydata$time/tau[j])   )^2  )/sigmay2
  }
 }
 out
}

pred.chi2.taufit2 = function(I0,tau) {
 out=matrix(nrow=length(I0),ncol=length(tau))
 for (i in 1:length(I0) ) {
  for (j in 1:length(tau) ) {
  xy=c(I0[i],tau[j])
  out[i,j]= (N-2) + t(xy-theta) %*% solve(V.theta)  %*% (xy-theta)
  }
 }
 out
```

```
}

nsigma=c(1,2,3)
x=seq(0.95,1.05,0.001)
y=seq(2.5,3.5,0.005)
# profilo effettivo in nero
contour(x,y, chi2.taufit2(x,y) ,  levels = (N-2) + nsigma^2,
 xlab=expression(I0),ylab=expression(tau))
# predizione approssimata in rosso
contour(x,y, pred.chi2.taufit2(x,y) ,  levels =(N-2) + nsigma^2,
col="red",add=T )
```

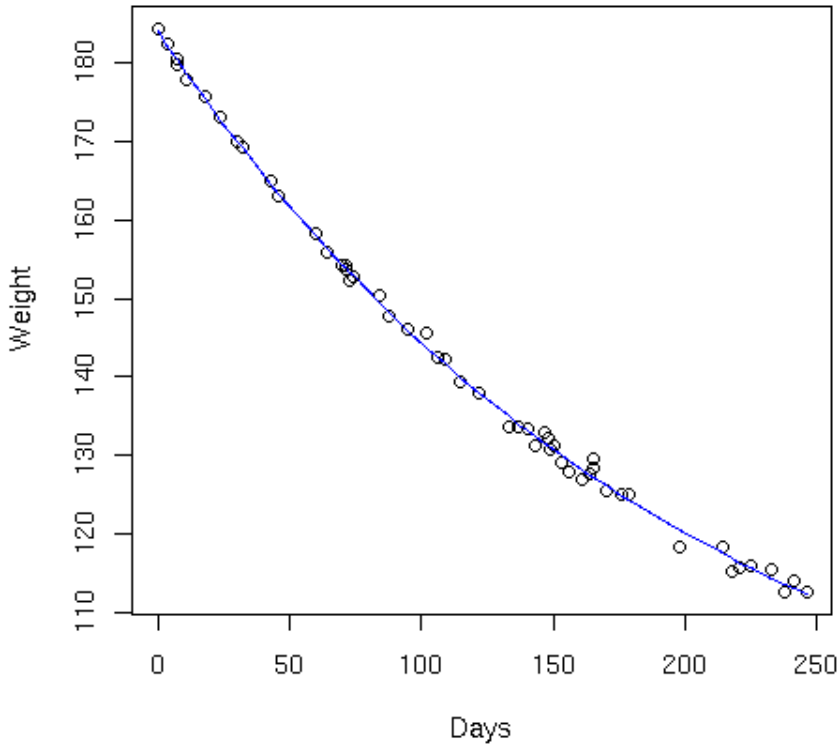


Esempio 4.3.2 *Analisi di una dieta*
Dall'analisi dei dati nel dataframe *wtloss* (pacchetto MASS), supponiamo che l'andamento del peso (variabile $Weight$, in Kg) di un paziente obeso sottoposto ad una dieta in funzione del tempo ($Days$, in giorni) sia

$$W = W_0 + W_E \cdot 2^{-D/D_{1/2}}$$

Vogliamo stimare i parametri W_0 (peso ideale), W_E (peso in eccesso), $D_{1/2}$ (tempo di dimezzamento del peso in eccesso), ed il parametro di correlazione lineare fra le stime di W_0 e W_E.
Essendo il modello non lineare, utilizziamo la funzione *nls()* con i tre parametri liberi

```
 library(MASS)
 plot(Weight ~ Days,data=wtloss)
 theta.start=c(W.0=100, W.e=100,D.half=200)
 fit=nls(Weight ~ W.0 + W.e * 2^(-Days/D.half) , data=wtloss, start=theta.start)
 print(summary(fit))
 lines(wtloss$Days,predict(fit),col="blue")

# matrice di correlazione fra i parametri
 Vcor = cov2cor(vcov(fit))
 rho = Vcor["W.0","W.e"]
 ngl=length(wtloss$Weight)-3
 drho =  (1-rho^2)/sqrt(ngl)
 cat("rho(W.0,W.e)=",round(rho,3)," +/- ",round(drho,3),"\n")
```



Da cui otteniamo

```
Parameters:
       Estimate Std. Error t value Pr(>|t|)
W.0      81.374      2.269   35.86   <2e-16 ***
```

```
W.e     102.684      2.083   49.30   <2e-16 ***
D.half  141.910      5.295   26.80   <2e-16 ***

rho(W.0,W.e)= -0.989  +/-  0.003
```

Test del modello

Sapendo che la bilancia ha una precisione di 0.5 Kg, testiamo ora la bontà del modello fittato.

Supponendo che la deviazione standard di ogni misura sia determinata dalla precisione della bilancia $\sigma = 0.5$ Kg, possiamo calcolare il χ^2 del fit ed eseguire un test dell'ipotesi che segua una distribuzione di Pearson con $N - 3$ gradi di libertà:

```
> chi2=deviance(fit)/0.5^2
> cat("chi2=",chi2," con ",ngl," g.d.l. : p-value=",
      1-pchisq(chi2,df=ngl),"\n")
chi2= 156.9788  con  49  g.d.l. : p-value= 3.100853e-13
```

Evidentemente l'errore nella misura non basta a spiegare la varianza dei residui del fit, che probabilmente è dominata da fluttuazioni fisiologiche del peso, e magari da qualche strappo alla dieta!

4.4 Fits di massima verosimiglianza

La soluzione più generale, ma anche più dispendiosa in termini di risorse di calcolo, per eseguire un *fit* di un campione di dati ad un modello parametrico, consiste nel trovare il massimo, usando le tecniche numeriche di ottimizzazione sopra descritte, della funzione di likelihood $\mathcal{L}(\mathcal{S};\underline{\theta})$ al variare dei parametri $\underline{\theta}$.
In questo modo possiamo trovare gli stimatori ML anche nel caso in cui gli errori non siano gaussiani e dunque il metodo dei minimi quadrati non sia strettamente giustificato.

Questo **fit di massima verosimiglianza** è utile anche nel caso in cui si voglia confrontare la distribuzione delle variabili osservate con un modello teorico, come nel caso del fit di un istogramma visto nel par. 4.2.7. Minimizzando la quantità

$$-\log \mathcal{L}(\mathcal{S};\underline{\theta}) = -\sum_i \log\left(d_y(y_i;\underline{\theta})\right) \tag{4.53}$$

si può risolvere il problema senza raggruppare le osservazioni in intervalli ("unbinned fit") e utilizzando dunque al meglio l'informazione disponibile. Se però vogliamo effettuare un test di bontà del fit, non possiamo in generale predire la statistica attesa di $(-\log\mathcal{L})_{min}$, a differenza del caso di χ^2_{min} che sappiamo seguire la distribuzione di Pearson. Per effettuare il test e calcolare il *p–value* è normalmente necessario fare ricorso ad una simulazione.

La funzione mle() nel pacchetto stats4 del software R è concepita per eseguire, utilizzando la funzione optim(), fits di massima verosimiglianza. L'errore sui parametri è stimato invertendo l'Hessiano di $-\log\mathcal{L}$.

Esempio 4.4.1 *Modello per i dati di un'eruzione*
La variabile *waiting* del dataframe $faithful$ descrive il tempo di attesa (in minuti) fra un'eruzione e l'altra di un geyger. Testiamo l'ipotesi che la sua PDF sia la somma di due gaussiane

$$f(w) = \alpha\frac{1}{\sqrt{2\pi}\sigma_1}e^{-(w-\mu_1)^2/2\sigma_1^2} + (1-\alpha)\frac{1}{\sqrt{2\pi}\sigma_2}e^{-(w-\mu_2)^2/2\sigma_2^2}$$

stimando i 5 parametri incogniti $\alpha, \mu_1, \sigma_1, \mu_2, \sigma_2$.

Il problema potrebbe essere risolto fittando l'istogramma della distribuzione della variabile *waiting* come visto nel par. 4.2.7. Pena un maggiore dispendio di CPU, possiamo effettuare un fit non "binnato", massimizzando direttamente la funzione di likelihood

$$L(\alpha, \mu_1, \sigma_1, \mu_2, \sigma_2) = \prod_i f(w_i; \alpha, \mu_1, \sigma_1, \mu_2, \sigma_2)$$

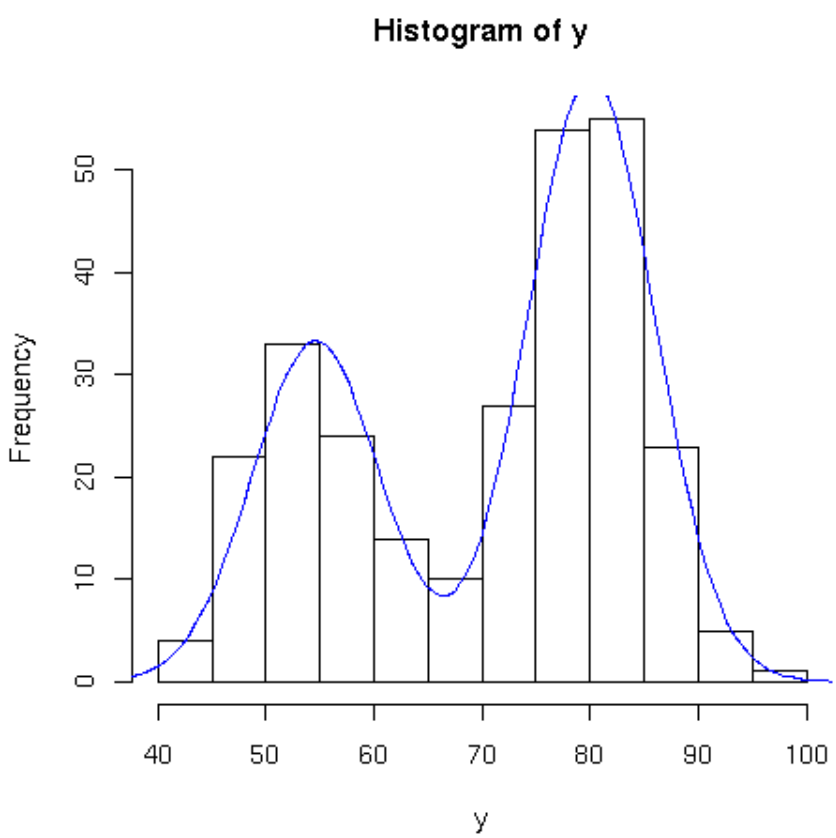
Usiamo per questo la funzione *mle*() del pacchetto *stats*4 che permette di minimizzare $-\log(L)$ con vari algoritmi di ottimizzazione. Scegliamo l'algoritmo L-BFGS-B, in cui devono essere specificati i limiti per i parametri del fit. Notiamo che nel nostro caso i parametri (μ_1, σ_1) e (μ_2, σ_2) possono facilmente essere "scambiati" (con un probabile fallimento della minimizzazione) se non rendiamo chiara la loro interpretazione scegliendo in modo sensato i valori di partenza e i limiti entro i quali eseguire la minimizzazione.

```
library(stats4)
geysermodel = function(x,alpha,mu1,sigma1,mu2,sigma2) {
alpha * dnorm(x,mean=mu1,sd=sigma1) +
   (1-alpha) * dnorm(x,mean=mu2,sd=sigma2)
}

geyser.fit = function(y=faithful$waiting) {
 h=hist(y)
 startp=
   list(alpha=0.5,mu1=55,sigma1=5 ,mu2=80,sigma2=5)
 minp=c(alpha=0.1,mu1=45,sigma1=1 ,mu2=75,sigma2=1)
 maxp=c(alpha=0.9,mu1=60,sigma1=10,mu2=85,sigma2=10)

 llgeyser = function(alpha,mu1,sigma1,mu2,sigma2) {
  -sum(log(geysermodel(y,alpha,mu1,sigma1,mu2,sigma2)) )
 }

# l'argomento di mle deve essere una funzione che ritorna -logL
 fit=mle(llgeyser,start=startp,lower=minp,upper=maxp,method="L-BFGS-B")
 print(summary(fit))
 cf=fit@coef
 curve(geysermodel(x,cf[1],cf[2],cf[3],cf[4],cf[5]) *
       (h$breaks[2]-h$breaks[1]) * length(y),
```

```
       add=T,col="blue")
 list( min= fit@min , coef= cf, var=vcov(fit))
}
```

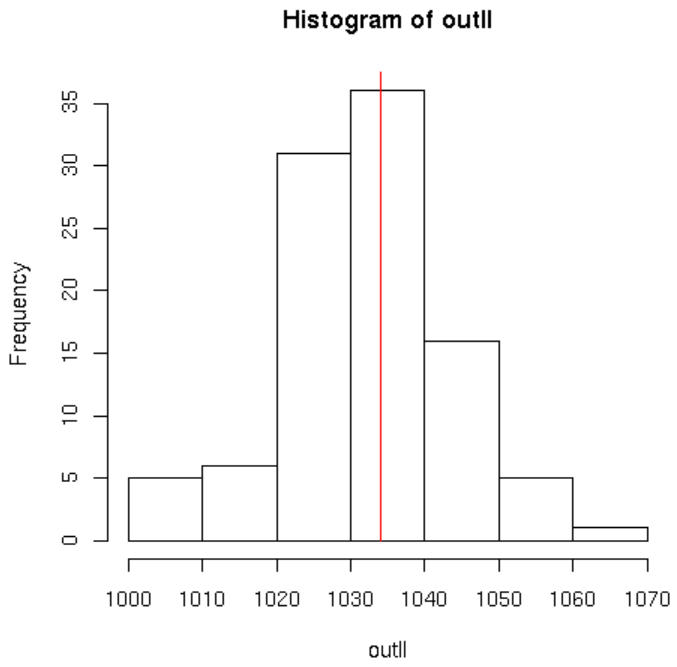
La funzione $geyser.fit()$ così implementata ritorna una lista con i valori e la matrice var./cov. dei parametri, e il minimo ottenuto di $-\log(L)$.

```
> expfit = geyser.fit()
Maximum likelihood estimation

Call:
mle(minuslogl = llgeyser, start = startp, method = "L-BFGS-B",
    lower = minp, upper = maxp)

Coefficients:
         Estimate Std. Error
alpha   0.3608887 0.03116472
mu1    54.6152983 0.69972712
sigma1  5.8715085 0.53739009
mu2    80.0912040 0.50459193
sigma2  5.8677196 0.40096112

-2 log L: 2068.004
```



Per poter eseguire un test di bontà del fit, è necessario conoscere la distribuzione attesa di $min(-\log(L))$ nell'ipotesi che il modello sia corretto.
Essendo un'impresa ardua determinare analiticamente questa distribuzione nel nostro caso (e questo vale per tutti i casi in cui la distribuzione $f(w)$ non sia particolarmente semplice), ricorriamo alla simulazione di $Nsim$ esperimenti con n valori della nostra variabile simulati secondo la funzione $f(w)$:

```
geyser.check = function(Nsim=100) {
# inizializziamo un vettore che conterra' i valori di min(log(L)) di
# ogni esperimento simulato
 outll=c()
 n=length(faithful$waiting)
 for (i in 1:Nsim) {
# il numero di eventi del primo tipo segue una distribuzione binomiale
# con prob. alpha
    n1=rbinom(1,size=n,prob=expfit$coef["alpha"])
    n2=n-n1
    sample=c( rnorm(n1,mean=expfit$coef["mu1"],sd=expfit$coef["sigma1"]) ,
              rnorm(n2,mean=expfit$coef["mu2"],sd=expfit$coef["sigma2"]) )
    simfit=geyser.fit(sample)
    outll[i]=simfit$min
 }
# distribuzione ottenuta dalla simulazione:
 hist(outll)
 cat("min logL =",expfit$min,"\n")
# disegniamo una linea verticale in corrispondenza del valore sperimentale
 lines(c(expfit$min,expfit$min),c(0,Nsim))
}
```



Come osserviamo dal grafico, il valore ottenuto sperimentalmente è perfettamente compatibile con il nostro modello.

Il metodo ML permette anche di tener conto nel fit del parametro "numero di eventi" N, che di solito segue una statistica di Poisson con valore atteso λ. Se λ è correlato con gli altri parametri, il nostro fit risulterà distorto a meno di non considerare la **funzione di likelihood estesa**

$$\mathcal{L}_e(\mathcal{S};\lambda,\underline{\theta}) = \frac{e^{-\lambda}\lambda^N}{N!}\prod_{i=1}^{N} \mathcal{d}_y(y_i;\underline{\theta}) \tag{4.54}$$

$$\log \mathcal{L}_e(\mathcal{S};\lambda,\underline{\theta}) = k - \lambda + \sum_i \log\left[\lambda \mathcal{d}_y(y_i;\underline{\theta})\right] \tag{4.55}$$

La funzione estesa è utile anche nel caso in cui si abbia un campione costituito dalla sovrapposizione di diverse componenti $\mathcal{d}_i$

$$\mathcal{d}(y) = \sum_j p_j \mathcal{d}_j(y) \tag{4.56}$$

e si vogliano ottenere dal fit le popolazioni $\lambda_j = p_j\lambda$ di ciascuna componente. In tal caso, il fit con la funzione estesa

$$\log \mathcal{L}_e(\mathcal{S}) = k - \sum_j \lambda_j + \sum_i \log\left[\sum_j \lambda_j \mathcal{d}_j(y_i)\right] \tag{4.57}$$

permette di ottenere direttamente le stime di λ_j con gli errori che tengono conto anche delle fluttuazioni globali di λ.

Capitolo 5

Incertezze sistematiche

Oltre che da incertezze "casuali" (o "incertezze statistiche"), che spiegano la variabilità dei risultati di una misura e che abbiamo affrontato nei capitoli precedenti, ogni processo di misura può essere affetto anche da incertezze, dette "sistematiche", che non variano al ripetersi della misurazione. Riprendendo dal cap. 1 il nostro primo esempio di misura di una massa su una bilancia, potremmo avere che, a causa della precisione finita della calibrazione dello strumento, tutte le nostre misure sono sistematicamente sovrastimate di una quantità s:

$$m_{exp} = m + \epsilon_{stat} + s \tag{5.1}$$

Al contrario dell'errore statistico ϵ_{stat}, che è una variabile aleatoria, s ha un valore ben definito che rimane costante al ripetersi della misura. L'incertezza consiste nella nostra ignoranza su questo valore.

La valutazione dell'errore sistematico è spesso la parte più insidiosa del lavoro di analisi dati. I metodi per stimare queste incertezze, e tenerne conto nel risultato di un'analisi statistica, sono introdotti in questo capitolo.

5.1 Modelli per le incertezze sistematiche

Abbiamo visto che la valutazione quantitativa dell'errore statistico su una misura necessita di un preciso criterio di interpretazione, in termini probabilistici, del risultato. Ad esempio, se riportiamo il valore osservato di uno stimatore gaussiano come 4.72 ± 0.13, sappiamo che abbiamo una probabilità del 68.3% che l'intervallo dato contenga il valore vero del parametro θ.

Vorremmo avere un criterio per valutare analogamente anche le incertezze sistematiche. Tuttavia in linea di principio, nell'ambito dell'interpretazione frequentista, non si potrebbe trattare un'incertezza sistematica con metodi statistici, per il semplice motivo che essa non è una variabile aleatoria e dunque non le si può associare una funzione di probabilità. Possiamo aggirare questo problema pensando l'errore sistematico s come un campionamento da una variabile aleatoria S che assume un valore diverso per ogni esperimento. Conoscendo la

distribuzione di probabilità $d(S)$ in questo "spazio degli esperimenti", possiamo procedere considerando s come un errore casuale, che è però correlato per tutte le misure del nostro singolo esperimento.
Riprendendo il nostro esempio di misura di massa 5.1, Supponiamo che il costruttore della bilancia abbia effettuato delle misure in base alle quali sappiamo che l'errore di calibrazione s segue una distribuzione normale con deviazione standard σ_{syst}. Noi disponiamo di una sola bilancia, ed s ha un valore fisso che non conosciamo, ma possiamo pensarlo come un valore casuale estratto secondo la distribuzione data. Sappiamo inoltre (e possiamo verificarlo facendo molte misure), che la risoluzione della bilancia è σ_{stat}, ovvero la variabile ϵ_{stat} segue una distribuzione normale con deviazione standard σ_{stat}.
Trattando m_{exp} come il valore misurato in uno dei possibili esperimenti (ovvero, tenendo conto che avremmo potuto farlo con un'altra bilancia), avremo

$$\sigma^2(m_{exp}) = \sigma^2(\epsilon_{stat}) + \sigma^2(s) = \sigma^2_{stat} + \sigma^2_{syst} \tag{5.2}$$

I due tipi di errore si sommano dunque in quadratura, poiché formalmente, nello "spazio dei possibili esperimenti", possiamo trattare gli errori sistematici come variabili aleatorie. Per lo stesso motivo, in presenza di molte sorgenti di errore sistematico, i vari contributi vengono sommati in quadratura (e dunque possiamo trascurare tutti gli errori che siano almeno un ordine di grandezza più piccoli di quello dominante) e, invocando il teorema del limite centrale, di solito si assume che l'errore risultante segua una distribuzione normale.
Bisogna tuttavia fare attenzione a considerare le correlazioni, introdotte dall'errore sistematico, fra due misure m_1 e m_2 fatte con la stessa bilancia. Se facciamo il valore atteso sullo spazio degli esperimenti in cui la bilancia è scelta casualmente (e dunque $E(m_1) = E(m_2) = m + E(s)$), ma tenendo conto che le due misure hanno sempre lo stesso bias s, avremo

$$\begin{aligned} cov(m_1, m_2) &= E\left[(m_1 - E(m_1))(m_2 - E(m_2))\right] = E\left[(\epsilon_1 + s - E(s))(\epsilon_2 + s - E(s))\right] = \\ &= E\left[(s - E(s))^2\right] = \sigma^2_{syst} \end{aligned} \tag{5.3}$$

dove si è usato il fatto che le variabili ϵ_1 e ϵ_2 sono a media nulla e indipendenti.

La matrice varianza–covarianza delle due variabili è dunque

$$V(m_1, m_2) = \begin{pmatrix} \sigma^2_{stat} + \sigma^2_{syst} & \sigma^2_{syst} \\ \sigma^2_{syst} & \sigma^2_{stat} + \sigma^2_{syst} \end{pmatrix} \tag{5.4}$$

che può essere facilmente estesa ad un numero qualsiasi di misure. Per la media su N misure $\overline{m} = \sum m_i/N$ avremo:

$$\sigma(\overline{m}) = \sqrt{\frac{\sigma^2_{stat}}{N} + \sigma^2_{syst}} \tag{5.5}$$

Se dunque l'errore statistico diminuisce come $1/\sqrt{N}$, la componente sistematica non dipende dalla statistica dell'esperimento e spesso costituisce il limite ultimo alla precisione della misura. L'eq. 5.5 ci dice che una statistica "ragionevole" per l'esperimento considerato è $N \sim \frac{\sigma^2_{stat}}{\sigma^2_{syst}}$, per

la quale i due contributi si equivalgono. Continuare ad acquisire dati molto oltre questo valore non produce alcun miglioramento nella precisione del risultato.

Esempio 5.1.1 *Misure con incertezza sistematica comune*
Vogliamo determinare gli errori statistico, sistematico e totale per la misura del peso lordo P_l di un vasetto di yogurt, sapendo che il peso netto P_n e la tara P_t sono misurati con la stessa bilancia, soggetta ad un errore sistematico δP =1 g, mentre l'errore casuale è gaussiano con σ_P =0.7 g.

I valori sperimentali P'_n e P'_t hanno una parte sistematica comune S:

$$P'_n = P_n + S \quad , \quad P'_t = P_t + S$$

Il peso lordo è dunque

$$P_l = P'_n + P'_t + 2S$$

da cui

$$\sigma^2(P_l) = \sigma^2(P'_n) + \sigma^2(P'_t) + 4\sigma^2(S) = 2\sigma_P^2 + 4(\delta P)^2$$

$(\sigma(P_l))_{stat} = \sqrt{2}\sigma_P = 1.0$g
$(\sigma(P_l))_{syst} = 2(\delta P) = 2.0$g
$(\sigma(P_l))_{tot} = 2.2$g
In modo equivalente, avremmo potuto scrivere la matrice varianza/covarianza di P_n e P_t

$$V = \begin{pmatrix} \sigma_P^2 + (\delta P)^2 & (\delta P)^2 \\ (\delta P)^2 & \sigma_P^2 + (\delta P)^2 \end{pmatrix}$$

e applicare la formula di propagazione degli errori

$$\sigma^2(P_l) = \sigma^2(P_n) + \sigma^2(P_t) + 2cov(P_n, P_t)$$

5.2 Stima di errori sistematici

Nella pratica, difficilmente si hanno a disposizione dati sperimentali (come quelli del costruttore di bilance) che ci permettano di dare un fondamento frequentista alla trattazione dell'incertezza sistematica. Più spesso, l'errore sistematico può solo essere stimato, in modo grossolano e sulla base del buon senso, quantificando un livello di confidenza sulla dimensione dei possibili effetti che possono distorcere la nostra misura. L'errore sistematico è infatti considerato una sorta di parametro di qualità della misura, che dipende dalla cura con cui gli effetti vengono valutati e tenuti sotto controllo dallo sperimentatore.
Bisogna dunque necessariamente adottare un approccio bayesiano. Ad esempio, supponiamo di disporre di due bilance, costruite indipendentemente, per effettuare la nostra misura di massa. Eseguendo una serie di misure con entrambe, appuriamo che i risultati dei due strumenti differiscono sistematicamente della quantità Δ. Naturalmente, nessuno ci garantisce che le due

bilance non siano entrambe soggette ad un errore sistematico molto superiore a Δ. Tuttavia, sapendo che i due strumenti sono stati calibrati nel modo migliore possibile per cercare di minimizzare l'errore sistematico residuo, è ragionevole, sulla base dell'informazione disponibile, assumere che l'errore sistematico delle nostre misure sia dell'ordine di Δ. La legge di probabilità bayesiana da associare a questo errore ha un margine di arbitrarietà: possiamo ad esempio scegliere una distribuzione uniforme nell'intervallo $[-\Delta, +\Delta]$, una distribuzione normale con $\sigma = \Delta$, etc. Assegnare una legge di probabilità bayesiana alla nostra ignoranza ci permette, come il buon senso suggerisce, di poter combinare in quadratura i diversi contributi sistematici indipendenti e l'errore statistico. Tuttavia, susistendo un margine di soggettività nella valutazione dell'errore sistematico, **è buona norma citare separatamente i due tipi di errore**. Se dunque dovessimo pubbliocare su una rivista scientifica il risultato della nostra misura di massa, scriveremmo qualcosa come

$$m = (5.67 \pm 0.08(\text{stat.}) \pm 0.10(\text{syst.}))\, g$$

Non vi sono regole rigorose per verificare la presenza o stimare la dimensione di effetti sistematici. Alcune procedure raccomandabili sono:

- lo studio della possibile dipendenza del risultato da qualunque variabile possa introdure un effetto (ad esempio tempo, temperatura, ...). L'analisi della varianza/covarianza costituisce lo strumento appropriato per testare l'effetto ed eventualmente quantificarlo;
- il confronto dei risultati ottenuti con strumenti e/o metodologie di analisi diversi;
- il confronto dei risultati ottenuti indipendentemente, sugli stessi dati, da diversi sperimentatori;
- il test della compatibilità dei dati con tutte le ipotesi su cui la misura è basata.

Esempio 5.2.1 *Misure di resistenza meccanica*
La tabella nel file
/afs/math.unifi.it/service/Rdsets/ceramic.rdata
contiene i risultati di misure di resistenza meccanica (in unità arbitrarie) di campioni di ceramica. Vogliamo determinare il valore medio di questa grandezza (variabile *Strength*), tenendo conto di eventuali effetti sistematici dovuti al laboratorio di test (variabile *Lab*, con 8 possibili valori) e dalla partita dei campioni (variabile *Batch*, 2 possibili valori).

Per testare la possibile dipendenza dalle variabili *Lab* e *Batch*, effettuiamo un'analisi della varianza con significatività del 5%. Se dunque il test di Fisher ha un *p–value* inferiore al 5%, diremo che c'è una evidenza di effetto sistematico. Per quanto riguarda *Lab*, il test è superato (anche se di poco):

```
> data = read.table("/afs/math.unifi.it/service/Rdsets/ceramic.rdata",skip=1)
> summary(aov(Strength ~ Lab, data=data))
             Df  Sum Sq Mean Sq F value  Pr(>F)
Lab           1   18332   18332  3.3065 0.06963 .
Residuals   478 2650115    5544
```

e non consideriamo dunque una possibile differenza sistematica fra i risultati di diversi laboratori di test. Per la variabile *Batch* c'è invece un effetto evidente:

```
> summary(aov(Strength ~ Batch, data=data))
             Df  Sum Sq Mean Sq F value    Pr(>F)
Batch         1  727138  727138  179.04 < 2.2e-16 ***
Residuals   478 1941309    4061
```

Una stima (grossolana) dell'effetto è

$$\sigma^2_{syst} \simeq \sigma^2(Strength) - \overline{\sigma^2(Strength|Batch = i)}$$

```
attach(data)
sigmatot=sd(Strength)
sigmastat=mean( tapply(Strength,Batch,sd)  )
sigmasyst=sqrt(sigmatot^2-sigmastat^2)
```

Da cui otteniamo $\sigma_{stat} \sim 65, \sigma_{syst} \sim 40$. Per il valor medio, ricordando la 5.5:

```
cat("media di Strength=",mean(Strength)," +/- ",
 sigmastat/sqrt(length(Strength))," (stat) +/- ", sigmasyst," (syst)\n")
```

da cui

$$\overline{Strength} = 650 \pm 3(stat.) \pm 40(syst.)$$

Alternativamente, e in modo altrettanto ragionevole quanto arbitrario, avremmo potuto considerare la differenza fra le medie dei due gruppi, pari a circa 78

```
> tapply(Strength,Batch,mean)
       1        2
688.9986 611.1560
```

e, assumendo una distribuzione uniforme dell'effetto sistematico fra ± 78, calcolare $\sigma_{syst} \sim (2 \cdot 78)/\sqrt{12} \sim 45$

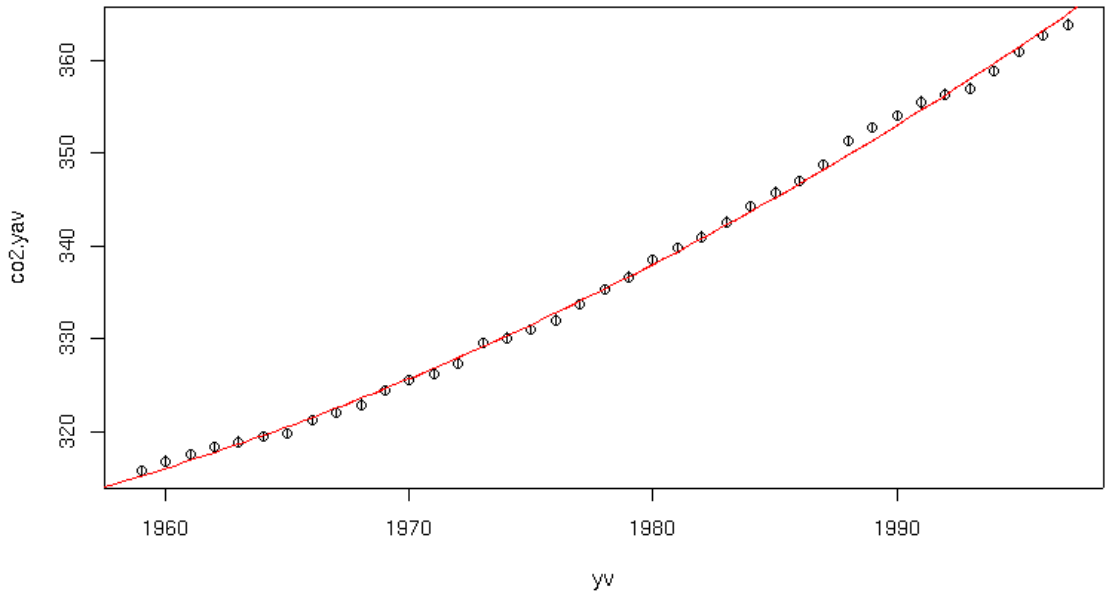
Errori sistematici nel risultato di un'analisi dati possono derivare anche da errori teorici nel modello usato per descrivere i dati. Se i dati non superano, ad esempio, un test di bontà del fit con il modello di dipendenza ipotizzato, questo potrà dipendere, oltre che da effetti sistematici nella misura, dall'inadeguatezza del nostro modello. Le predizioni di tale modello saranno allora affette da un'incertezza sistematica.

Esempio 5.2.2 *Modello di evoluzione della concentrazione atmosferica di CO2*
I valori tabulati nel file
/afs/math.unifi.it/service/Rdsets/maunaloa.rdata
mostrano l'evoluzione nel tempo della concentrazione di CO2 misurata presso il vulcano Mauna

Loa sulle isole Hawaii fra il 1959 e il 1997. Vogliamo parametrizzare questi dati assumendo una dipendenza quadratica dall'anno oltre ad una dipendenza stagionale, in modo da predire, sulla base di questo modello, i valori per gli anni successivi, stimando l'errore statistico e quello sistematico sulla previsione.

A prima vista, i dati mostrano un ragionevole accordo con la dipendenza quadratica dall'anno, che corrisponde ad assumere un aumento lineare nel tempo delle emissioni. Cominciamo ad eseguire un fit sul valore medio annuale di concentrazione c in funzione dell'anno:

```
maunaloa=read.table("/afs/math.unifi.it/service/Rdsets/maunaloa.rdata")
c=maunaloa$co2
y=maunaloa$year
m=maunaloa$month
time=y+(m-.5)/12

year.fit = function() {
# calcoliamo il valor medio, col suo errore, per ciascun anno
 yv = as.numeric( tapply(y,y,mean) )
 co2.yav =as.numeric( tapply(c,y,mean))
 co2.sd = as.numeric( tapply(c,y,sd))
 co2.n = as.numeric( tapply(c,y,length))
 co2.sigma = co2.sd/sqrt(co2.n)
# grafico dei dati, con relativi errori
 plot(co2.yav ~ yv)
 np = length(co2.yav)
 symbols(yv,co2.yav,rectangles=matrix(ncol=2,c(rep(0,np),2*co2.sigma)),
         add=T,inches=FALSE)
# fit nell'ipotesi di andamento quadratico
 fit0 = lm(co2.yav ~ yv + I(yv^2), weights=1/co2.sigma^2 )
 cf=coef(fit0)
 curve(cf[1]+cf[2]*x+cf[3]*x^2,add=T,col="red")
 chi2 = deviance(fit0)
 dof=np-3
 cat("chi2= ",chi2," dof=",dof,"   p-value is ",1-pchisq(chi2,df=dof),"\n")
 fit0
}
> year.fit()
chi2=  52.95874  dof= 36    p-value is  0.03391743

Call:
lm(formula = co2.yav ~ yv + I(yv^2), weights = 1/co2.sigma^2)

Coefficients:
(Intercept)           yv      I(yv^2)
  4.909e+04   -5.061e+01    1.312e-02
```



Il piccolo valore di *p–value* (3.4%) del test χ^2 dimostra una deviazione sistematica dal modello di cui dovremo tener conto.

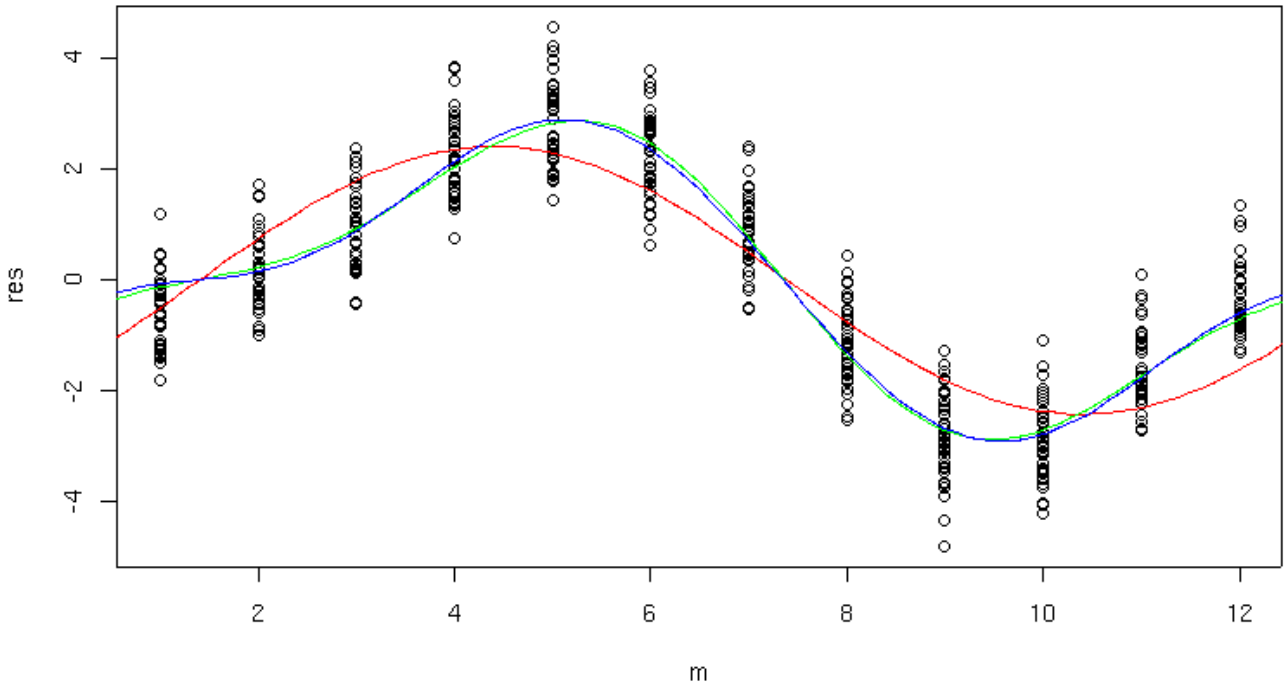
Mettendo in grafico i residui del fit in funzione del mese, vediamo che la varianza σ_w^2 è dominata dalle variazioni stagionali, che possiamo cercare di parametrizzare empiricamente con la funzione periodica

$$g(m) = \sum_{j=1}^{l} A_j sin\left(2\pi j \frac{(m-\phi)}{12}\right)$$

dove m è la variabile mese e A_j, ϕ sono i parametri liberi del fit. Essendo il modello non lineare, usiamo stavolta la funzione *nls()*:

```
month.fit = function() {
 fit0=year.fit()
 c.pred = predict(fit0,newdata=data.frame(yv=y))
 res = c-c.pred
 plot(res ~ m)
 startp=c(fs=4,phase=0)
 mfit1=nls(res ~ fs*sin(2*pi*(m-phase)/12),start=startp)
 cf1=coef(mfit1)
 curve(cf1[1]*sin(2*pi*(x-cf1[2])/12),add=T,col="red")

 startp2=c(cf1,fs2=0)
 mfit2=nls(res ~ fs*sin(2*pi*(m-phase)/12)+fs2*sin(2*pi*2*(m-phase)/12),start=startp2
 cf2=coef(mfit2)
 curve(cf2[1]*sin(2*pi*(x-cf2[2])/12)+cf2[3]*sin(2*pi*2*(x-cf2[2])/12),add=T,col="gre
 print(anova(mfit1,mfit2))

 startp3=c(startp2,fs3=0)
```

```
 mfit3=nls(res ~ fs*sin(2*pi*(m-phase)/12)+fs2*sin(2*pi*2*(m-phase)/12)+
      fs3*sin(2*pi*3*(m-phase)/12), start=startp3)
 cf3=coef(mfit3)
 curve(cf3[1]*sin(2*pi*(x-cf3[2])/12)+cf3[3]*sin(2*pi*2*(x-cf3[2])/12)+
       cf3[4]*sin(2*pi*3*(x-cf3[2])/12),add=T,col="blue")
 print(anova(mfit2,mfit3))
 mfit3
}
```

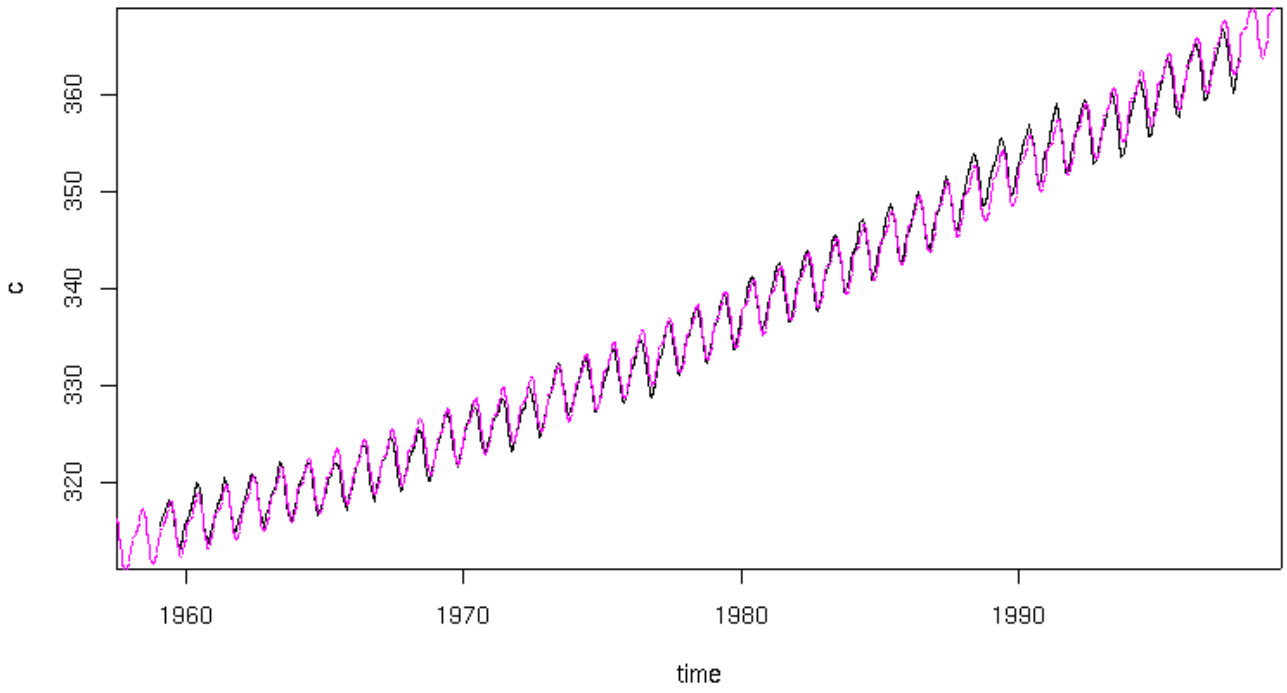


Otteniamo un accordo soddisfacente con $l = 3$, che non migliora aggiungendo ulteriori armoniche. I residui di questo fit sono ragionevolmente descritti da una distribuzione normale, e possiamo dunque interpretarli come un errore casuale sulla misura.
Mettiamo ora insieme il modello, fittando i dati in funzione del tempo con 7 parametri (3 per l'aumento quadratico e 4 per la variazione stagionale):

```
mymodel.month= function(x,phase,m1,m2=0,m3=0) {
 m1 * sin(2*pi* (x-phase)/12) +
 m2 * sin(2*pi* 2* (x-phase)/12) +
 m3 * sin(2*pi* 3* (x-phase)/12)
}

mymodel = function(year, month, th) {
# qui year deve essere anno-1950
 th[1] + th[2]*year + th[3] * year^2 +
  mymodel.month(month,th[4],th[5],th[6],th[7])
}
```

```
full.fit = function() {
 startp=c(t1=1000,t2=0,t3=0,ts1=2.,phase=1.3,ts2=-1,ts3=0)
 fittone=nls(c ~ mymodel(y-1950,m,c(t1,t2,t3,ts1,phase,ts2,ts3)),
,start=startp)
 print(summary(fittone))
 resid.error=sqrt(deviance(fittone)/df.residual(fittone))
 stat.error = mean(tapply(resid(fittone), y, sd) )
 syst.error = sqrt(resid.error^2 - stat.error^2)
 cat("resid. error=",resid.error,"   stat. error=",stat.error,
     "   syst. error=",syst.error,"\n")
 fittone
}
> fittone = full.fit()
> plot(c ~ time, type='l')
> curve(mymodel(as.integer(x-1950),12*(x-1950-as.integer(x-1950)),coef(fittone)),
        col="magenta",add=T,n=1000)
```

(si è considerata la variabile $(year - 1950)$ anzichè $year$, in modo che l'ordine di grandezza dei coefficienti non differisca troppo, per evitare che errori di approssimazione facciano fallire l'ottimizzazione numerica del fit)



L'errore residuo del fit è 0.74, ma diventa ~ 0.36 se consideriamo le variazioni all'interno di ciascun anno. Possiamo allora pensare che la predizione del nostro modello è affetta da un'incertezza sistematica dipendente dall'anno (ma comune a tutti i valori di un dato anno) che stimiamo come

$\sigma_\delta \sim \sqrt{0.74^2 - 0.36^2} \sim 0.6$.

Riassumendo, disponiamo di un modello a 7 parametri

$$\begin{aligned}\lambda(year, month) =& \theta_1 + \theta_2(year - 1950) + \theta_3(year - 1950)^2 + \\ &+ \theta_4 \sin(2\pi(month - \theta_5)/12) + \theta_6 \sin(2 \cdot 2\pi(month - \theta_5)/12) \\ &+ \theta_7 \sin(2 \cdot 3\pi(month - \theta_5)/12)\end{aligned}$$

che ci permette di riprodurre l'andamento temporale dei dati con un errore casuale residuo $\sigma_{stat} \sim 0.36$ e un residuo sistematico, dipendente dall'anno, $\sigma_\delta \sim 0.6$.

Naturalmente potremmo cercare di ridurre l'effetto sistematico residuo aggiungendo altri parametri al fit in funzione dell'anno, ma, non disponendo di un adeguato modello teorico per descrivere le variazioni residue, il modello non sarebbe più predittivo. Basiamo allora su questo modello le nostre predizioni sul valore di $co2$ atteso ad un generico tempo $T = (y', m')$

$$co2(y', m') = \lambda(y', m') + \epsilon(\sigma_{stat}) + \Delta(y'; \sigma_\delta)$$

che saranno soggette ad un errore sistematico Δ dovuto alle imperfezioni del modello.

Calcoliamo ora la predizione, ed il relativo errore, per gli anni fra il 1998 e il 2008, in modo da poterli confrontare con le misure sperimentali che si possono trovare nel file
/afs/math.unifi.it/service/Rdsets/maunaloaRecent.rdata.
L'errore statistico su $\lambda(T)$ è ottenuto dalla formula di propagazione degli errori

$$\sigma(\lambda(T))^2 \simeq \sum_{i=1}^{7} \left(\frac{\partial\lambda}{\partial\theta_i}(T)\right)^2 \sigma(\theta_i)^2 + \sum_{i,j} \left(\frac{\partial\lambda}{\partial\theta_i}(T)\right)\left(\frac{\partial\lambda}{\partial\theta_i}(T)\right) cov(\theta_i, \theta_j) = AV_\theta A^T$$

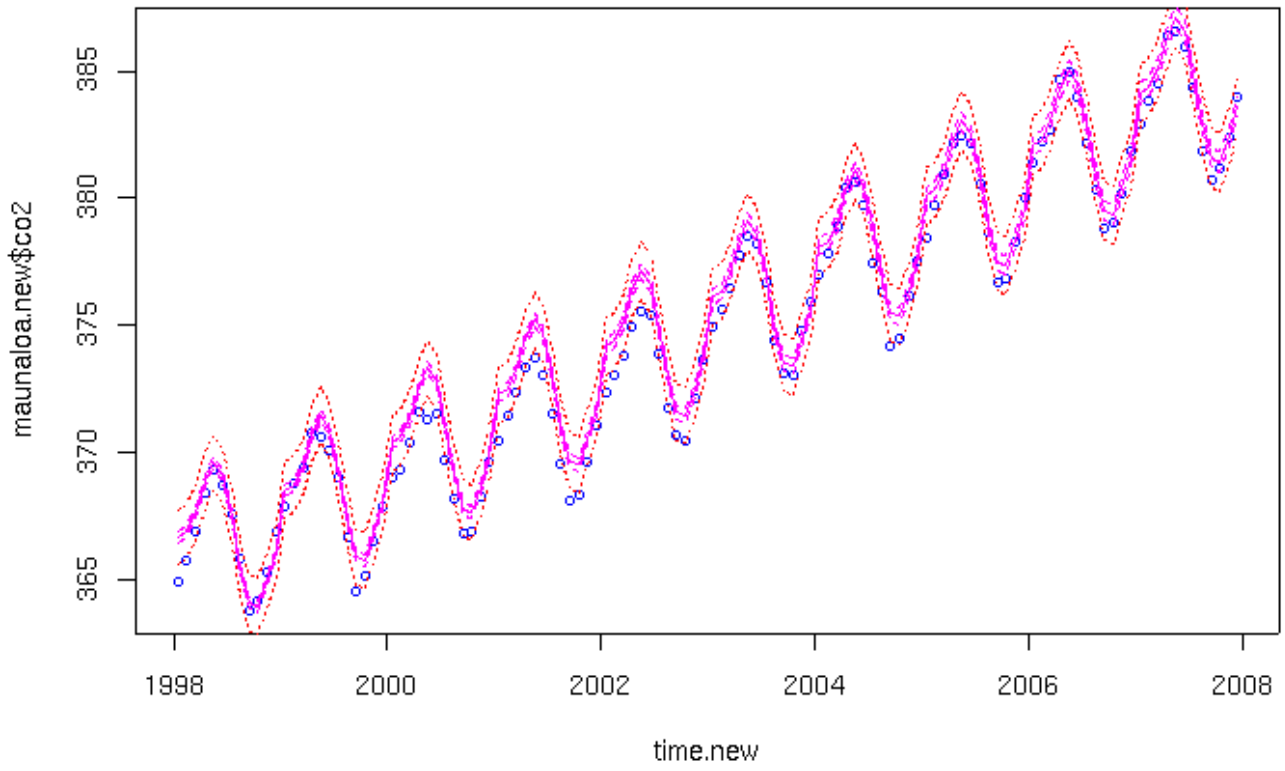
dove la matrice A è la matrice del modello per i valori di T per i quali vogliamo la predizione

$$A_{ij} = \frac{\partial\lambda}{\partial\theta_j}(T_i)$$

A questo errore bisognerà sommare l'errore sistematico. Ci aspettiamo infatti che tutti i punti misurati in un anno possano differire sistematicamente dal nostro modello di una quantità dell'ordine di σ_δ La somma dei due errori è fatta solitamente in quadratura, anche se è buona norma citare separatamente i due errori.

```
mymodel.deriv = function( year, month, th) {
# calcolo della matrice con le derivate
 A=matrix(nrow = length(year), ncol= 7)
 for (i in 1:length(year)) {
  y=year[i]
  m=month[i]
  A[i,1]=1
  A[i,2]=y
  A[i,3]=y^2
  A[i,4]=sin(2*pi* (m-th[5])/12)
```

```
  A[i,5]=-th[4] * cos(2*pi*(m-th[5])/12) * 2*pi/12
         -th[6] * cos(2*pi*2*(m-th[5])/12) * 2*pi*2/12
         -th[7] * cos(2*pi*3*(m-th[5])/12) * 2*pi*3/12
  A[i,6]=sin(2*pi* 2* (m-th[5])/12)
  A[i,7]=sin(2*pi* 3* (m-th[5])/12)
 }
 A
}

mymodel.predict = function(conflevel = 0.9) {
 fittone=full.fit()
 resid.error=sqrt(deviance(fittone)/df.residual(fittone))
 stat.error = mean(tapply(resid(fittone), y, sd) )
 syst.error = sqrt(resid.error^2 - stat.error^2)

 maunaloa.new=read.table("/afs/math.unifi.it/service/Rdsets/maunaloaRecent.rdata",
                         header=T)

 pred.best = mymodel(maunaloa.new$year-1950,maunaloa.new$month, coef(fittone))
 pred.A = mymodel.deriv(maunaloa.new$year-1950,maunaloa.new$month, coef(fittone))
 pred.sigma = sqrt ( diag (pred.A %*% vcov(fittone) %*% t(pred.A)) )

# calcoliamo l'intervallo di confidenza per l'incerteza statistica
 nsigma = qt((1+conflevel)/2,df=df.residual(fittone))
 pred.min=pred.best - nsigma *  pred.sigma
 pred.max=pred.best + nsigma *  pred.sigma

 time.new=maunaloa.new$year+(maunaloa.new$month-0.5)/12

 plot(time.new,maunaloa.new$co2,col="blue",cex=.5)

 lines(time.new, pred.best, col="magenta")
 lines(time.new, pred.min, col="magenta", lty=2)
 lines(time.new, pred.max, col="magenta", lty=2)

# aggiungiamo l'incertezza sistematica, stimata da syst.error
 pred.syst.min=pred.best - nsigma *  sqrt(pred.sigma^2 + syst.error^2)
 pred.syst.max=pred.best + nsigma *  sqrt(pred.sigma^2 + syst.error^2)

 lines(time.new, pred.syst.min, col="red", lty=3)
 lines(time.new, pred.syst.max, col="red", lty=3)

}
```



Verifichiamo dunque che i valori misurati dopo il 1997 sono del tutto compatibili con l'andamento previsto in base ai dati precedenti, tenendo conto delle incertezze sistematiche sul modello.

Bibliografia

[1] Brandt, S. *Statistical and Computational Methods in Data Analysis.* North-Holland, 1976.

[2] Broyden, C. G. The convergence of a class of double-rank minimization algorithms. I: General considerations. 76–90.

[3] Cowan, G. *Statistical Data Analysis.* Clarendon Press, Oxford, 1997.

[4] Eadie, W. T., Drijard, D., James, F. E., Roos, M., and Sadoulet, B. *Statistical Methods in Experimental Physics.* North-Holland, Amsterdam, 1971.

[5] Fisher, R. A. On the mathematical foundations of theoretical statistics. *Philos. Trans. R. Soc. Lond. A 22* (1922), 133–142.

[6] Freeman, J. A., and Skapura, D. M. *Neural Networks.* Addison-Wesley, Reading, MA, USA, 1991.

[7] Gosset, W. S. The probable error of a mean. *Biometrika 6*, 1 (March 1908), 1–25. Originally published under the pseudonym "Student".

[8] Kolmogorov, A. *Grundbegriffe der Warscheinlichkeitsrechnung.* Springer, Berlin, 1933. Translated into English by Nathan Morrison (1950), Foundations of the Theory of Probability, Chelsea, New York. Second English edition 1956.

[9] Maindonald, J., and Braun, J. *Data Analysis and Graphics Using R: An Example-based Approach.* Cambridge University Press, New York, NY, USA, 2006.

[10] Neyman, J. *Phil. Trans. A236* (1937), 333.

[11] Pearson, K. On a criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. *Philosophy Magazine 50 (5)* (1900), 157–175.

[12] Venables, W. N., Smith, D. M., and others. *An Introduction to R.* http://cran.r-project.org/doc/manuals/R-intro.pdf.