

# Lezione n. 9

## 9.1 Rappresentazioni grafiche di verosimiglianze bivariate

Prima di procedere alla determinazione numerica del punto di massimo della funzione di (log) verosimiglianza, possiamo fornirne una rappresentazione grafica; in questo modo possiamo avere un'idea della forma della funzione ed eventualmente anche del punto iniziale da cui far partire l'algoritmo di massimizzazione. Molte famiglie parametriche sono espresse in funzione di due argomenti (ad es. famiglie con parametri di locazione e scala); in questo caso non è possibile ricorrere al comando `plot`, ma in R possiamo ricorrere a due funzioni, `contour` e `persp`, delle quali la prima fornisce il grafico delle curve di livello di una funzione bivariata, mentre la seconda restituisce una vera e propria rappresentazione tridimensionale della nostra funzione. Vedremo esclusivamente l'uso della funzione `contour`; la struttura degli argomenti principali è la stessa per le due funzioni; per le specifiche aggiuntive della funzione `persp` si rimanda all'`help` (consultabile scrivendo sulla riga di comando `help(persp)`).

ESEMPIO 9.1 [*Verosimiglianza dei parametri di una Gamma (cfr. Esercizio 8.2)*]

A puro scopo esemplificativo, supponiamo di aver estratto un campione di numerosità  $n = 50$  da una Gamma con parametri  $\alpha = 2$  e  $\lambda = 1.4$ : scriviamo dunque `x.dat_rgamma(50,2,rate=1.4)`.

Per ottenere una rappresentazione grafica della verosimiglianza di  $\alpha, \lambda$  alla luce del campione generato, utilizziamo il comando `contour`, che traccia le linee di livello della funzione da rappresentare.

Nel caso bivariato, sia che si utilizzi `contour`, sia che si utilizzi `persp`, bisogna considerare due sequenze di valori, una per ciascun parametro, e fornire poi al programma una *matrice* contenente i valori che la funzione da rappresentare assume su una *griglia* di punti, griglia che è costituita dall'insieme di tutte le possibili coppie di valori che sono stati assegnati ai parametri.

A questo scopo, notiamo che R contiene la funzione `outer`, che calcola il prodotto esterno: se  $X$  e  $Y$  sono due vettori, il prodotto esterno  $A$  di  $X$  e  $Y$  è una matrice di dimensione  $c(\text{length}(X), \text{length}(Y))$  i cui elementi sono  $A[\text{index}.x, \text{index}.y] = \text{FUN}(X[\text{index}.x], Y[\text{index}.y], \dots)$ . In sostanza ad ogni coppia di elementi viene applicata la funzione `FUN`, che va passata tra gli argomenti della funzione `outer`. `FUN` deve essere una funzione con almeno due argomenti (associati a  $x$  e  $y$  rispettivamente) che opera sui vettori *elemento per elemento*. Ad esempio:

```
x <- 1:3
y <- 1:3
outer(x,y, '+')
```

restituisce la matrice  $3 * 3$  di elemento  $i, j$  pari a  $i+j$ :

```
      [,1] [,2] [,3]
[1,]    2    3    4
[2,]    3    4    5
[3,]    4    5    6
```

Per poterla utilizzare con la funzione `outer`, scriviamo la verosimiglianza in una funzione `R` di *tre* argomenti: i due parametri, considerati come argomenti *separati*, più il vettore dei dati.

```
gamma.llik_function(a,l,dati)
{ sommalog <- sum(log(dati))
  somma <- sum(dati)
  n <- length(dati)
  ll <- n*a*log(1)-n*log(gamma(a))+(a-1)*somialog-l*somma
  return(ll)}
```

Confrontiamo questa funzione con quella dell'Esercizio 8.2, in cui i parametri della verosimiglianza erano stati passati come *un unico parametro vettoriale*. Si noti che per massimizzare la verosimiglianza è necessario scrivere una funzione in cui i parametri su cui massimizzare la funzione sono espressi tramite un unico argomento (vettoriale), mentre per rappresentare graficamente la verosimiglianza abbiamo bisogno di -almeno- due argomenti separati, uno per parametro. Pertanto in generale è necessario modificare la nostra funzione che restituisce la verosimiglianza a seconda dell'uso che ne vogliamo fare.

Al solito, è necessario scegliere un rettangolo su cui rappresentare la funzione: scegliamo l'intervallo  $(0, 4]$  per  $\alpha$  e l'intervallo  $(0, 6]$  per  $\lambda$ . Scriviamo

```
a <- seq(0.01,4,length=50)
l <- seq(0.01,6,length=50)
z <- outer(a,l, gamma.llik, dati=x.dat)
```

Per  $z$  otteniamo una matrice di dimensione  $50 * 50$  i cui elementi rappresentano i valori della log-verosimiglianza in corrispondenza di ciascuna coppia di valori dei parametri  $a, l$ .

```
contour(a, l,z,nlevels=300)
title("log verosimiglianza")
#per disegnare la curva tridimensionale:
persp(a, l, z, theta = 30, phi = 30, expand = 0.8, col = "lightblue",
      ltheta = 120, shade = 0.75, ticktype = "detailed",
      xlab = "a", ylab = "l", zlab = "lv")
title("log verosimiglianza")
```

`nlevels` consente di specificare quante linee di livello vanno disegnate. In alternativa è possibile specificare il livello a cui effettuare i tagli della funzione: ad esempio

```
lev <- seq(min(z),max(z),7.5)
contour(a, l,z,levels=lev)
```

Dai grafici si rileva immediatamente la forma di paraboloide rovesciato della funzione di log verosimiglianza. Il suo punto di massimo  $(\hat{\alpha}, \hat{\lambda})$  va ricavato numericamente, ma dato che il campione `x.dat` è stato simulato da una  $\text{Gamma}(2,1.4)$  possiamo intanto rappresentare sul grafico il parametro “vero” della distribuzione: `points(c(2,1.4),col=3)`. A seconda del campione ottenuto e della sua numerosità, la zona intorno al massimo della verosimiglianza sarà più o meno vicina al parametro “vero”. Una volta ricavata la stima di massima verosimiglianza, è anche possibile confrontare quest’ultima con il valore che essa stima.

**Esercizio 9.1.** *[Segue da Esempio 9.1]*

Disegnare la verosimiglianza dei parametri del modello gamma avendo osservato il campione utilizzato nell’esercizio 8.2 (e contenuto nel file `surv.txt`).

## 9.2 Intervalli di confidenza

È bene sottolineare che l’intervallo di confidenza non costituisce un’affermazione probabilistica sull’appartenenza del parametro ad un intervallo prefissato; si tratta piuttosto

di un intervallo *casuale* i cui estremi dipendono dal campione (e sono quindi variabili con esso); a seconda del particolare campione estratto, tale intervallo può o meno coprire il parametro, che è una quantità fissata, pur essendo incognita. La nozione di livello di confidenza  $(1-\alpha)$  indica con che probabilità, ripetendo l'operazione di campionamento, l'intervallo va a coprire il parametro.

L'esempio 9.2 utilizza il metodo di costruzione di intervalli basato su pivot. Un pivot per il parametro  $\theta$  è una variabile aleatoria  $Q$  la cui distribuzione non dipende da  $\theta$ ; si noti che  $Q$  può dipendere da  $\theta$ , ed in questo un pivot si differenzia da una statistica, che non dipende dal parametro (altrimenti non sarebbe numericamente determinabile), avendo invece una distribuzione che dipende dal parametro.

L'esempio 9.3 chiarisce la natura e il significato della stima per intervallo. Fissato un valore del parametro, simulando il principio del campionamento ripetuto, generiamo  $m$  campioni dalla stessa popolazione e costruiamo per ciascuno di essi un intervallo di confidenza. Alcuni di essi conterranno il parametro da stimare, altri non lo conterranno; se  $m$  è sufficientemente elevato, calcolando la percentuale di intervalli che non contengono il valore prefissato del parametro (o la percentuale di campioni estratti che danno luogo a tali intervalli), otterremo una stima di  $\alpha$ . Fissando ad esempio  $\alpha = 0.05$ , dobbiamo attenderci che circa il 5% degli intervalli di confidenza costruiti non coprano il valore vero del parametro. Naturalmente nella pratica si costruisce uno soltanto di questi intervalli; non essendo possibile stabilire se l'informazione fornita da tale intervallo è corretta o meno, si ragiona in media sullo spazio dei possibili campioni, lavorando sul livello di confidenza  $\alpha$ .

**ESEMPIO 9.2** [*Intervallo di confidenza per la varianza di una normale*]

Siano  $X_i$   $i = 1, \dots, n$   $n = 14$  variabili aleatorie i.i.d.  $\sim N(\mu, \sigma^2)$ . Determinare un intervallo di confidenza al livello  $1 - \alpha = 0.95$  per il parametro  $\sigma^2$ .

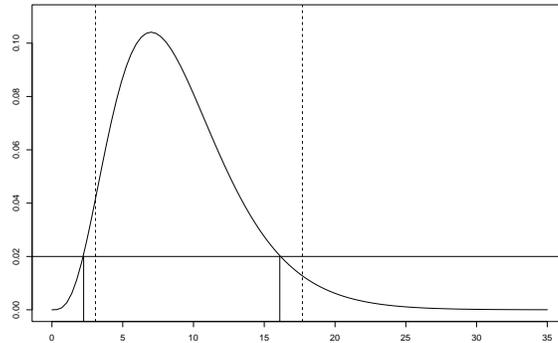
**SOLUZIONE.** *Un pivot per il parametro  $\sigma^2$  si ricava immediatamente essere  $Q = (n-1)S^2/\sigma^2$  che è noto seguire una legge chi quadrato con  $n - 1$  gradi di libertà (indipendentemente da  $\sigma$ ). Possiamo pertanto determinare un intervallo  $q_1, q_2$  in cui il pivot  $Q$  è contenuto con probabilità  $1 - \alpha$ :*

$$1 - \alpha = \Pr \left\{ q_1 < \frac{(n-1)S^2}{\sigma^2} < q_2 \right\} \quad (9.1)$$

*Dato che  $\frac{(n-1)S^2}{\sigma^2} \sim \chi_{n-1}^2$ , possiamo determinare  $q_1$  e  $q_2$  in base ai quantili di una legge  $\chi_{n-1}^2$ . In particolare possiamo scegliere  $q_1 : \int_0^{q_1} f_Q(u)du = \alpha/2$  e  $q_2 : \int_0^{q_2} f_Q(u)du = 1 - \alpha/2$ , valori che possono essere determinati scrivendo*

```
q1 <-qchisq(0.025, 13)
```

```
q2 <-qchisq(0.975, 13)
```



Si osservi che ci sono molti modi per determinare due valori  $q_1, q_2$  in modo che  $\Pr\{q_1 < Q < q_2\} = 1 - \alpha$ ; in questo caso si è scelto di determinare i quantili che lasciano rispettivamente alla propria sinistra e alla propria destra due aree uguali e pari ad  $\alpha/2$ ; una soluzione alternativa, che conduce a un intervallo più piccolo se la distribuzione non è simmetrica consiste nel tagliare orizzontalmente la densità di  $Q$  determinando i valori  $q_1, q_2$  in modo che  $f(q_1) = f(q_2)$  e  $\Pr\{q_1 < Q < q_2\} = 1 - \alpha$ . La figura evidenzia la differenza tra il metodo che assegna la stessa probabilità alle code e il metodo appena menzionato. Nel caso in cui la densità sia simmetrica, i due metodi coincidono, ma in generale, come si può notare dalla figura, il metodo basato sull'assegnazione della stessa probabilità alle due code conduce a intervalli più ampi. Quest'ultima procedura è preferita nelle applicazioni perchè di implementabilità immediata, mentre l'altra necessita di procedure numeriche per l'individuazione dei valori  $q_1, q_2$ . Nella pratica, inoltre, non vi sono grandi differenze di ampiezza tra gli intervalli così costruiti. Si noti che nel nostro esempio, data la convergenza del chi quadrato alla normale, la distribuzione di riferimento tende a diventare sempre più simmetrica al crescere di  $n$ , per cui effettivamente le differenze possono essere di qualche rilievo solo per campioni molto piccoli. "Invertendo" quindi l'intervallo (9.1), usando i quantili del chi quadrato (dove  $\chi_{\alpha, \nu}^2$  rappresenta il quantile di livello  $\alpha$  della distribuzione Chi quadrato con  $\nu$  g.d.l.)

$$\left\{ \chi_{\alpha/2, n-1}^2 < \frac{(n-1)S^2}{\sigma^2} < \chi_{1-\alpha/2, n-1}^2 \right\}$$

otteniamo

$$\left\{ \frac{\chi_{\alpha/2, n-1}^2}{(n-1)S^2} < \frac{1}{\sigma^2} < \frac{\chi_{1-\alpha/2, n-1}^2}{(n-1)S^2} \right\}$$

da cui si ricava che

$$\left\{ \frac{(n-1)S^2}{\chi_{1-\alpha/2, n-1}^2} \right\} < \sigma^2 < \left\{ \frac{(n-1)S^2}{\chi_{\alpha/2, n-1}^2} \right\} = \left\{ \frac{(n-1)S^2}{q_2} < \sigma^2 < \frac{(n-1)S^2}{q_1} \right\}$$

è un intervallo di confidenza per  $\sigma^2$  al livello  $1 - \alpha$ .

A scopo esemplificativo, estraiamo un campione da una legge  $N(0, 4)$ :

```
dati <- rnorm(21,0,2).
```

La funzione che determina una stima per intervallo del parametro  $\sigma$  al livello  $1 - \alpha$  è la seguente:

```
intconf <-function (dati, alpha=0.05)
{
n <- length(dati)
s <- var(dati)
q1 <- qchisq(alpha/2,n-1)
q2 <- qchisq(1-alpha/2,n-1)
a <- (n-1)*s/q2
b <- (n-1)*s/q1
return(c(a,b))
}
```

(naturalmente la funzione restituisce un estremo inferiore osservato e un estremo superiore osservato, ossia due realizzazioni di due variabili casuali). ■

ESEMPIO 9.3 [Natura e significato della stima per intervallo]

Costruiamo una rappresentazione grafica che permette di apprezzare il significato dell'intervallo di confidenza e al contempo determiniamo un'approssimazione del livello di confidenza  $1 - \alpha$ .

Simuliamo dalla stessa popolazione una serie di  $m$  campioni indipendenti e costruiamo per ciascuno di essi la corrispondente stima per intervallo di  $\sigma^2$ . Per ciascun campione possiamo applicare la funzione `intconf`, che restituisce gli estremi inferiore e superiore dell'intervallo di confidenza realizzato in corrispondenza di un dato campione. Rappresentiamo questo intervallo su un grafico (comando `segments`) e registriamo se l'intervallo realizzato copre o meno il valore del parametro  $\sigma^2$  che abbiamo fissato nella simulazione. In questo modo possiamo incrementare un contatore che registra in quanti degli intervalli simulati il parametro vero non è stato coperto dall'intervallo. La frequenza relativa di tali intervalli, se  $m$  è sufficientemente elevato, rappresenta una stima di  $\alpha$ . Allo

stesso tempo possiamo rappresentare in rosso sul grafico gli intervalli di confidenza che non coprono il valore vero del parametro. Per  $m$  elevato la rappresentazione grafica si presenta confusa; si suggerisce pertanto di utilizzare inizialmente un valore di  $m$  contenuto (ad es. 500), per osservare il grafico, e successivamente di richiamare la funzione con un  $m$  più elevato, specificando `plot=F`, per ottenere una stima più attendibile di  $\alpha$ .

In alternativa, se si richiama più volte la funzione con gli stessi parametri di popolazione (`media` e `sqm`) e con  $m$  contenuto, data l'indipendenza tra i campioni è possibile utilizzare le stime di  $\alpha$  ottenute di volta in volta, facendone una media pesata con i valori di  $m$  utilizzati. In questo modo è possibile mantenere l'output grafico.

Il comando `segments` richiede che si inseriscano le (4) coordinate degli estremi del segmento da disegnare. Come tutti i comandi grafici, all'interno della funzione possiamo specificare ulteriori parametri grafici, ad esempio `col`.

```
sim.int<-function(m=1000,n=14,media=0,sqm=1,plot=T,alpha=0.05)
{
q <- qchisq(alpha/2,n-1)
am_((n-1)*(sqm^2))/q
err <- 0
if(plot==T){
plot(1:m,rep(sqm^2,m),ylim=c(0,am+2.5*am),type='l', ylab="ic")}
for (i in 1:m)
{dat <- rnorm(n,media,sqm)
res <- intconf(dat)
colore <- ifelse(res[2]<sqm^2,1,0)+ifelse(res[1]>sqm^2,1,0)
err <- err+colore
if(plot==T){
segments(i,res[1],i,res[2],col=colore+1)}
}
err <- err/m
if(plot==T){
text(round(m/2),2,round(err,3))}
return(err)
}
```

### 9.3 Test del rapporto di verosimiglianza generalizzato

Non sempre per il problema di ipotesi che si affronta è possibile individuare una statistica test con proprietà ottimali. In queste situazioni è possibile adottare il test del rapporto delle verosimiglianze generalizzato

$$\lambda = \frac{\sup_{\Theta_0} l(\theta)}{\sup_{\Theta} l(\theta)}.$$

Grazie ai risultati asintotici sulla distribuzione della trasformata di Wilks  $-2 \log \lambda$ , la procedura è applicabile a un gran numero di problemi di verifica di ipotesi parametriche; inoltre i test basati sul rapporto delle verosimiglianze generalizzato sono caratterizzati da proprietà generalmente soddisfacenti, e nei casi in cui si può costruire la statistica test ottimale, essa risulta equivalente a quella derivata con il rapporto delle verosimiglianze generalizzato.

**Esercizio 9.2.** [*Test sulla media di una normale*]

Si osserva il seguente campione di 12 osservazioni estratte da una popolazione normale con media incognita e varianza nota e pari a  $\sigma^2 = 1.2$ :

$$\underline{x} = (3.9, 3.8, 1.4, 3.5, 4.4, 0.8, 4.3, 3.1, 1.1, 3.8, 2.6, 0.9)$$

Sulla base dei dati osservati, si vuole verificare l'ipotesi  $H_0 : \mu = 0$  contro l'ipotesi  $H_1 : \mu \neq 0$  al livello  $\alpha = 0.05$ .

Costruire il test al livello  $\alpha=0.05$  basato sul RVG e determinare la sua funzione di potenza. Rappresentarla graficamente.