Università degli Studi di Padova
Dipartimento di Scienze Statistiche
Corso di Laurea Triennale in
Statistica per le Tecnologie e le Scienze

Relazione Finale

Intervalli di confidenza per il parametro di sovradispersione con dati di conteggio

Relatore Prof. Alessandra Salvan
Dipartimento di Scienze Statistiche

Laureanda Virginia Skerl
Matricola 2045570

Anno Accademico 2023/2024

Indice

Introduzione **1**

1 Modelli per la sovradispersione **3**
1.1 Sovradispersione nei modelli lineari generalizzati . . . 3
1.2 Sovradispersione nei dati di conteggio . . . 5
1.3 Modello di regressione binomiale negativa . . . 6
1.4 Modello di regressione binomiale negativa lineare . . . 8
1.5 Modello di quasi-verosimiglianza . . . 8

2 Intervalli di confidenza per il parametro di sovradispersione **11**
2.1 Intervalli di confidenza . . . 11
2.2 Intervalli di confidenza di Wald . . . 12
2.3 Intervalli di confidenza profilo . . . 13

3 Intervalli di confidenza tramite bootstrap **15**
3.1 Introduzione . . . 15
3.2 Modelli lineari generalizzati e residui . . . 15
3.3 Metodi basati sul bootstrap . . . 18
3.3.1 Bootstrap semi-parametrico . . . 18
3.3.2 Bootstrap non parametrico . . . 20
3.4 Tipi di intervalli bootstrap . . . 20

4 Esempi di applicazione **23**
4.1 Introduzione . . . 23
4.2 Esempio 1: *cloth* . . . 23
4.3 Esempio 2: *bigcity* . . . 27

5 Simulazioni **31**
5.1 Coperture degli intervalli di confidenza . . . 31
5.2 Simulazioni con $\phi = 1$ . . . 31
5.3 Simulazioni con $\phi > 1$ . . . 32

Conclusioni **33**

Appendice **37**

Bibliografia **49**

Introduzione

La presente relazione riguarda intervalli di confidenza per il parametro di sovradispersione in modelli per dati di conteggio. La sovradispersione è un fenomeno che si verifica quando la varianza della variabile risposta, valutata senza utilizzare tutte le assunzioni del modello, è maggiore rispetto a quella prevista dal modello. Allo stesso modo, è possibile che la varianza sia inferiore a quella prevista, in tal caso si parla di sottodispersione. Nel seguito ci si concentrerà solamente sugli aspetti di sovradispersione in quanto più diffusi.

Nel contesto dei modelli lineari generalizzati risulta particolarmente restrittiva l'assunzione sulla varianza imposta dal modello di Poisson per dati di conteggio; esso prevede che il valore atteso e la varianza della variabile risposta siano equivalenti. Per tale motivo ci si concentrerà sulla sovradispersione nei dati di conteggio.

La sovradispersione provoca in genere una sottostima della varianza degli stimatori dei coefficienti di regressione e conseguentemente degli standard error, portando così alla scorretta selezione dei parametri da comprendere nel modello. La stima e la costruzione di intervalli di confidenza per il parametro di sovradispersione risultano, dunque, di interesse. In particolare, in modelli parametrici è possibile utilizzare metodi di verosimiglianza (Salvan et al., 2020, Capitolo 5). Al contrario, se si considera un approccio semi-parametrico come quello di quasi-verosimiglianza, l'unica possibilità sembra quella di ricorrere a metodi di ricampionamento.

La relazione è organizzata come segue: nel Capitolo 1 si presentano i modelli tipicamente utilizzati per tener conto della sovradispersione. Nel Capitolo 2 e nel Capitolo 3 sono introdotti rispettivamente l'approccio parametrico e non parametrico per la costruzione di intervalli di confidenza. Nel Capitolo 4 si mostrano delle applicazioni a casi di studio in cui è presente sovradispersione. Infine, nel Capitolo 5 si effettuano alcune simulazioni per diversi valori del parametro di sovradispersione ed alcune considerazioni sulla copertura dei rispettivi intervalli di confidenza.

Capitolo 1

Modelli per la sovradispersione

1.1 Sovradispersione nei modelli lineari generalizzati

I modelli lineari generalizzati permettono di estendere i modelli di regressione lineare normale a modelli con distribuzione della risposta diversa dalla normale e introducendo la funzione di legame che lega il valore atteso della risposta e predittore lineare.

Siano $Y_1, \ldots, Y_n$ variabili casuali indipendenti. Si assume che la distribuzione della risposta Y_i sia parte della classe della famiglia di dispersione esponenziale, un'ampia classe a cui appartengono alcuni esempi notevoli di distribuzioni sia univariate che multivariate, quali la distribuzione normale, gamma, Poisson, binomiale, esponenziale, normale multivariata e multinomiale. In particolare, $Y_i,\ i = 1, \ldots, n$, appartiene alla classe di famiglie di dispersione esponenziale univariate se ha densità esprimibile nella forma

$$p(y_i; \theta_i, \phi) = \exp\left\{\frac{\theta_i\, y_i - b(\theta_i)}{a_i(\phi)} + c(y_i, \phi)\right\}, \tag{1.1}$$

dove $y_i \in S \subseteq \mathbb{R}$ con S supporto di Y_i , $\theta_i \in \Theta \subseteq \mathbb{R}$ con Θ spazio parametrico, $a_i(\phi) > 0$. Il parametro θ è detto naturale e ϕ di dispersione. Da tale espressione generale, specificando le funzioni $a_i(\cdot),\ b_i(\cdot)$ e $c(\cdot)$, si ottengono diversi modelli parametrici.

Siano $\boldsymbol{x}_i$ vettori riga p-dimensionali delle variabili esplicative per l'i-esima osservazione ($i = 1, \ldots, n$), tali che vadano a comporre la matrice X di dimensione $p \times n$, e β il vettore colonna p-dimensionale dei coefficienti di regressione. Allora il predittore lineare è definito come $\eta = X\beta$ con componenti $\eta_i = \boldsymbol{x}_i\ \beta$. Infine, sia μ_i il valore atteso della variabile risposta Y_i, la funzione legame è la funzione $g(\cdot)$ nota, derivabile e monotona

crescente tale che

$$g(\mu_i) = \eta_i.$$

La funzione $g(\cdot)$ permette il collegamento tra valore atteso e predittore lineare. Tuttavia, è necessario che $g(\cdot)$ risolva il problema di coerenza in cui si incorre quando lo spazio delle medie M è sottoinsieme di $\mathbb{R}$, ossia sia tale che $g\colon M \to \mathbb{R}$.

Se Y_i ha densità (1.1), $E(Y_i) = \mu_i = b'(\theta_i)$ e $Var(Y_i) = a_i(\phi)\ b''(\theta_i)$. Inoltre, tramite la riparametrizzazione (μ_i, ϕ) definita da $\mu_i = \mu(\theta_i)$, con inversa $\theta_i = \theta(\mu_i)$, si ha

$$v(\mu_i) = b''(\theta_i)\big|_{\theta_i = \theta(\mu_i)}$$

tale che

$$Var(Y_i) = a_i(\phi)\ v(\mu_i), \tag{1.2}$$

dove la funzione $v(\mu_i)$ è detta funzione di varianza. Entro la classe delle famiglie di dispersione esponenziale univariate, la funzione di varianza caratterizza, assieme a M, il modello parametrico specifico, per la dimostrazione si veda Pace & Salvan (1996, Teorema 5.2).

La notazione compatta della classe delle famiglie di dispersione esponenziale univariate è

$$Y_i \sim DE_1(\mu_i, a_i(\phi)v(\mu_i)), \quad \text{con } \mu_i = g(\eta_i)^{-1} = g(\boldsymbol{x}_i\ \beta)^{-1}.$$

Quindi i parametri d'interesse nei modelli lineari generalizzati sono (β, ϕ) oppure solamente β nel caso in cui ϕ sia noto, questo è il caso per le famiglie notevoli i cui elementi caratterizzanti sono definiti. Nella Tabella 1.1 sono raccolti alcuni esempi di famiglie DE_1 con i relativi elementi caratterizzanti tra cui anche i parametri di dispersione. Dunque l'ipotesi sulla varianza della risposta prevista è specificata dall'equazione (1.2) e la sovradispersione è presente quando essa non viene rispettata dalla effettiva varianza della risposta, come si può riscontrare empiricamente.

TABELLA 1.1: Alcuni elementi caratterizzanti delle principali famiglie di dispersione esponenziale univariate.

	$N(\mu_i, \sigma^2)$	$Ga(\alpha, \frac{\alpha}{\mu_i})$	$Po(\mu_i)$	$\frac{1}{m_i}Bin(m_i, \mu_i)$
ϕ	σ^2	$\frac{1}{\alpha}$	1	1
$a_i(\phi)$	σ^2	$\frac{1}{\alpha}$	1	$\frac{1}{m_i}$
M	$\mathbb{R}$	$(0, +\infty)$	$(0, +\infty)$	(0,1)
$v(\mu_i)$	1	μ_i^2	μ_i	$\mu_i(1-\mu_i)$

La sovradispersione è un fenomeno comune per le distribuzioni la cui varianza è legata alla media come la distribuzione Poisson e quella binomiale, mentre si presenta raramente nella distribuzione normale dove la varianza è descritta dal parametro σ^2, non legato a μ_i. Nei prossimi paragrafi si affronta il caso di presenza di sovradispersione nei dati di conteggio.

L'adattamento dei modelli lineari generalizzati è possibile tramite il software R, sono implementati nella libreria `stats` che contiene la funzione `glm`, questa ha come argomenti importanti la formulazione del modello e la famiglia di appartenenza dei dati. Una volta specificata la famiglia, è possibile definire la funzione di legame `link` desiderata, se non fornita il programma ricorre alla funzione di default per la famiglia. Per i dati di conteggio la famiglia di riferimento è `poisson` che assume una funzione di legame logaritmica di default.

Per ulteriori approfondimenti riguardanti i modelli lineari generalizzati si rimanda a Salvan et al. (2020).

1.2 Sovradispersione nei dati di conteggio

I dati di conteggio si possono facilmente riscontrare in diversi ambiti di applicazione: dato un vettore riga p-dimensionale di variabili concomitanti $\boldsymbol{x}_i$ per l'i-esima osservazione allora y_i è il numero di eventi osservati per tale osservazione, $i = 1, \dots, n$. Si noti che $S = \{0, 1, \dots\}$, per cui $\sum_{i=1}^{n} y_i$ non è fissata a priori ma dipende dall'insieme di osservazioni considerato.

Per dati di conteggio con totale non prefissato il modello statistico di riferimento è quello di Poisson

$$Y_i \sim Po(\mu_i), \quad i = 1, \dots, n.$$

Tale distribuzione rientra nella classe di forma (1.1) con

$$a_i(\phi) = 1, \ b(\theta_i) = e^{\theta_i} \ \text{ e } \ c(y_i, \phi) = -\log(y_i!).$$

La distribuzione di Poisson assume funzione di varianza $v(\mu_i) = \mu_i$, dunque

$$Y_i \sim DE_1(\mu_i, \mu_i) \quad \text{con } E(Y_i) = Var(Y_i) = \mu_i = g(\eta_i)^{-1},$$

dove la funzione di legame è tipicamente quella logaritmica $g(\mu_i) = \log(\mu_i) = \eta_i$.

L'uguaglianza tra valore atteso e varianza della variabile risposta è spesso non rispettata causando così sovradispersione, in particolare spesso la variabilità è superiore alla media. Un test per saggiare l'adeguatezza del modello di Poisson contro l'alternativa di

sovradispersione è la statistica di Pearson

$$X^2 = \sum_{i=1}^{n} \frac{(y_i - \hat{\mu}_i)^2}{\hat{\mu}_i}$$

con $\hat{\mu}_i = g(\boldsymbol{x}_i\, \hat{\beta})^{-1}$, questa si distribuisce approssimativamente, sotto ipotesi del modello di Poisson, seguendo una χ^2_{n-p} quando $\hat{\mu}_i > 3$ (si veda ad esempio Rao & Chakravarti, 1956, paragrafo 1.1).

Se le ipotesi sottostanti al modello di Poisson non sono soddisfatte, può essere conseguenza dell'omissione dal modello di variabili esplicative o di termini di interazione rilevanti, della presenza di outliers e della scorretta specificazione della funzione di legame, in tal caso il problema della sovradispersione è risolto controllando la specificazione del modello e l'eventuale necessità di inserire ulteriori termini. In caso contrario, in cui l'alta variabilità sia intrinseca ai dati e la sovradispersione reale, si ricorre all'utilizzo di modelli più flessibili che permettono una variabilità superiore di quella prevista dai modelli lineari generalizzati.

Per dettagli riguardanti diverse cause e soluzioni della sovradispersione nei modelli Poisson si rinvia ai paragrafi 7.1 e 7.3 di Hilbe (2011), mentre il paragrafo 7.4 riporta altri test per la verifica della presenza di sovradispersione.

1.3 Modello di regressione binomiale negativa

Un approccio parametrico di estensione dei modelli lineari generalizzati per dati che presentano sovradispersione sono i modelli mistura. Posta $Y_i,\ i = 1, \ldots, n$, con densità $p_{Y_i}(y_i, \theta_i)$ dove $\theta_i \in \Theta$, si può introdurre eterogeneità ipotizzando che θ_i sia, a sua volta, realizzazione di una variabile casuale con densità $p_{\theta_i}(\theta_i, \psi_i)$, il parametro ψ_i è chiamato iperparametro.

I modelli mistura sono anche detti modelli gerarchici in quanto si può interpretare y_i come generato da un esperimento che si svolge in più stadi. Infatti, inizialmente si genera θ_i da $p_{\theta_i}(\theta_i, \psi_i)$, per poi generare y_i da $p_{Y_i}(y_i, \theta_i)$ che è interpretabile come densità condizionata. Tuttavia, essendo θ_i non osservabile, l'inferenza si basa sulla distribuzione marginale di Y_i e la densità del modello mistura risulta essere

$$p_{Y_i}(y_i, \psi_i) = \int_{\Theta} p_{\theta_i, Y_i}(\theta_i, y_i, \psi_i)\, d\,\theta_i = \int_{\Theta} p_{\theta_i}(\theta_i, \psi_i)\ p_{Y_i}(y_i, \theta_i)\, d\,\theta_i.$$

Per dati di conteggio si ipotizza $Y_i |\, \lambda_i \sim Po(\mu_i\, \lambda_i)$, dove $\mu_i > 0$ è fissato e $\lambda_i > 0,\ i =$

$1, \ldots, n$, ha distribuzione tale per cui

$$E(\lambda_i) = 1, \quad Var(\lambda_i) = \tau \geq 0.$$

In questo modo $E(Y_i) = \mu_i$ e $Var(Y_i) = E(Var(Y_i|\lambda_i)) + Var(E(Y_i|\lambda_i)) = \mu_i\,(1+\tau\mu_i)$, e dunque per $\tau = 0$ si ha una distribuzione Poisson e per $\tau > 0$ un'estensione della Poisson con variabilità maggiore. Utilizzando la notazione introdotta per i modelli mistura λ_i corrisponde a θ_i, mentre τ all'iperparametro ψ_i, uguale per ogni i.

I requisiti imposti su λ_i sono rispettati dalla distribuzione $Ga(\kappa, \kappa)$ con $\kappa = 1/\tau > 0$, sotto tale assunzione la densità del modello mistura risulta essere

$$\begin{aligned} p_(\, y_i; \lambda_i, \kappa) &= \int_0^{+\infty} \frac{\exp\{-\mu_i\,\lambda_i\}\,(\mu_i\,\lambda_i)^{y_i}}{y_i!}\,\frac{\kappa^\kappa \lambda_i^{\kappa-1} exp\{-\kappa, \lambda_i\}}{\Gamma(\kappa)}\,d\,\lambda_i \\ &= \frac{\Gamma(y_i+\kappa)}{y_i!\,\Gamma(\kappa)}\left(\frac{\mu_i}{\kappa+\mu_i}\right)^{y_i}\left(\frac{\kappa}{\kappa+\mu_i}\right)^{\kappa}. \end{aligned}$$

Posto $\pi_i = 1 + \kappa/\mu_i$ probabilità di successo, con κ intero, $Y_i + \kappa$ è la variabile casuale che rappresenta il numero di prove fino al κ-esimo successo, questa è detta binomiale negativa

$$Y_i + \kappa \sim Bineg(\kappa, \pi_i),$$

per cui Y_i ha distribuzione binomiale negativa traslata con $E(Y_i) = \mu_i$ e $Var(Y_i) = \mu_i\,(1 + \mu_i/\kappa)$.

Date le variabili concomitanti $\boldsymbol{x}_i$, il modello di regressione con risposta binomiale negativa assume y_i realizzazione di Y_i, $i = 1, \ldots, n$, con $\kappa = 1/\tau$ e $\mu_i = g(\boldsymbol{x}_i\ \beta)^{-1}$ dove $g(\cdot) = \log(\cdot)$. Con κ non noto, la distribuzione binomiale negativa non rientra nei modelli lineari generalizzati, tuttavia dallo studio della funzione di log-verosimiglianza $l(\beta, \tau)$ risulta che le equazioni di verosimiglianza per β hanno forma che coincide con la forma nei modelli lineari generalizzati. Inoltre, i parametri β e τ sono ortogonali. Di conseguenza, i loro stimatori di massima verosimiglianza sono asintoticamente indipendenti e risulta sufficiente disporre del blocco dell'informazione osservata o attesa relativo al parametro d'interesse.

Per l'adattamento di modelli con risposta binomiale negativa in R si utilizza più comunemente la funzione `glm.nb` della libreria `MASS`, con argomenti equivalenti a `glm`, ma in cui non è necessario specificare la famiglia. Oltre a questa esistono altre librerie e funzioni che risultano esserne estensioni e forniscono informazioni aggiuntive, come la funzione `vglm` del pacchetto `VGAM` che adatta modelli lineari generalizzati vettoriali e, dunque, permette di specificare diversi predittori lineari, tanti quanti il numero di

parametri del modello. Oppure la funzione `brnb` della libreria `brglm2` che calcola le stime anche con metodi, alternativi alla massima verosimiglianza, che riducono la distorsione dello stimatore.

1.4 Modello di regressione binomiale negativa lineare

Il modello di regressione binomiale negativa lineare prende il nome dalla forma lineare della varianza $Var(Y_i) = \tau\,\mu_i\,(1+\tau)$. Si tratta di un modello mistura dove $Y_i \sim Po(\lambda_i)$ e $\lambda_i \sim Ga(\mu_i,\, 1/\tau)$ da cui

$$\begin{gathered} E(Y_i) = \tau\,\mu_i = \tau\, g(\boldsymbol{x}_i\ \beta)^{-1} \quad \text{con } g(\cdot) = \log(\cdot), \\ Var(Y_i) = E(Y_i)\,(1+\tau) = E(Y_i)\,\phi. \end{gathered}$$

Si noti che il rapporto tra valore atteso e varianza è costante all'interno del modello, al contrario del modello di regressione binomiale negativa.

L'adattamento in R del modello di regressione NB1 può avvenire con l'utilizzo della funzione `gamlss` dell'omonimo pacchetto tramite l'opzione `family = NBII`, essa fornisce stima e standard error per il parametro τ, `sigma`, in scala logaritmica.

Per approfondimenti sull'argomento si consulti Hilbe (2011, paragrafo 10.2).

1.5 Modello di quasi-verosimiglianza

Un approccio alternativo a quello dei modelli mistura è quello semi-parametrico dei modelli di quasi-verosimiglianza, un modello statistico che si basa sulle ipotesi del secondo ordine dei modelli lineari generalizzati

$$\begin{gathered} E(Y_i) = g(\boldsymbol{x}_i\ \beta)^{-1}, \\ Var(Y_i) = \phi\, v(\mu_i) \quad \text{dove } \phi > 0 \text{ ignoto}, \\ Y_i \text{ e } Y_j \text{ indipendenti se } i \neq j\,. \end{gathered}$$

Esso è adatto a trattare risposte $Y_i,\ i = 1, \ldots, n$, sia continue che discrete, in particolare per risposte binomiali e Poisson permette un incremento di flessibilità rispetto alla specificazione parametrica dei modelli lineari generalizzati. Infatti, assumendo $\phi > 0$ ignoto, il modello consente una variabilità diversa rispetto alla forma (1.2) in cui il parametro di dispersione è noto e la varianza della risposta è interamente definita. Di conseguenza, il modello di quasi-verosimiglianza risulta essere un valido metodo

per tener conto della sovradispersione nei modelli binomiale e Poisson in cui le ipotesi imposte sulla varianza risultano spesso violate.

L'inferenza sul modello di quasi-verosimiglianza, basata su equazioni di stima non distorte, permette di dimostrare che nei modelli binomiali e Poisson la stima di β, e quindi del valore atteso della risposta, è equivalente a quella dei modelli lineari generalizzati corrispondenti (posto $\phi = 1$). Mentre la varianza stimata della risposta è moltiplicata per un fattore pari a $\tilde{\phi}$, stima di ϕ, solitamente basata sul metodo dei momenti

$$\tilde{\phi} = \frac{1}{n-p} \sum_{i=1}^{n} \frac{(y_i - \hat{\mu}_i)^2}{v(\hat{\mu}_i)} \quad \text{con } \hat{\mu}_i = g(\boldsymbol{x}_i\ \hat{\beta})^{-1}.$$

Questa stima, infatti, rimane consistente anche sotto le ipotesi del modello di quasi-verosimiglianza.

Per ulteriori dettagli sull'inferenza nei modelli di quasi-verosimiglianza si rinvia a Agresti (2015, paragrafi 8.1 e 8.3).

Infine, si noti che se la funzione di varianza rispetta l'assunzione del modello Poisson $v(\mu_i) = \mu_i$, il modello di quasi-verosimiglianza ha $Var(Y_i) = \phi, \mu_i$ di forma lineare, equivalente alla varianza del modello NB1 con $\phi = 1 + \tau$.

I modelli di quasi-verosimiglianza sono implementati nella libreria `stats`, `quasipoisson` e `quasibinomial` sono delle famiglie comprese nella funzione `glm` che prevede, rispettivamente, funzione di legame di default la funzione logit e quella logaritmica.

Capitolo 2

Intervalli di confidenza per il parametro di sovradispersione

2.1 Intervalli di confidenza

Una stima intervallare è una procedura inferenziale che incorpora la stima di un parametro con una misura d'incertezza, andando così a comporre un intervallo di confidenza $IC_{1-\alpha}$: un intervallo casuale tale che includa il vero valore del parametro d'interesse con una probabilità pari a $1-\alpha$, detto livello di confidenza, con α fissato a priori.

Gli intervalli di confidenza sono usualmente costruiti sulla base di una quantità pivotale, una funzione $q\colon S \times \Theta \to \mathbb{R}$ tale che, per ogni $\theta \in \Theta$, $q(Y,\theta)$ è variabile casuale con distribuzione di probabilità che non dipende da θ.

In molti casi non è disponibile una quantità esattamente pivotale per un parametro d'interesse, per cui si ricorre a quantità pivotali approssimate basate sulla verosimiglianza profilo.

Assumendo che il parametro d'interesse sia il parametro di dispersione ϕ, è possibile definire la log-verosimiglianza profilo $l_P(\phi)$ dalla verosimiglianza globale $L(\beta,\phi)$ come

$$l_P(\phi) = \log(L_P(\phi)) = \log(L(\hat{\beta}_\phi, \phi)),$$

dove $\hat{\beta}_\phi$ è la stima di massima verosimiglianza di β ottenuta con ϕ fissato. Lo stimatore di massima verosimiglianza di ϕ massimizza $l_P(\phi)$.

In seguito si presentano alcuni esempi notevoli di quantità pivotali e i relativi intervalli di confidenza utilizzati nell'inferenza statistica di un parametro scalare, come quello di dispersione.

2.2 Intervalli di confidenza di Wald

La quantità di Wald profilo è una quantità approssimativamente pivotale definita come

$$r_{eP}(\phi) = \frac{\hat{\phi} - \phi}{\sqrt{\widehat{Var}(\hat{\phi})}},$$

essa rappresenta una standardizzazione dello stimatore di massima verosimiglianza ottenuta valutando la distanza tra stima e vero valore, e tenendo conto dell'errore standard.

L'intervallo di confidenza costruito sulla base di tale quantità è l'insieme di valori

$$\left\{\phi > 0 \colon \mid r_{eP}(\phi) \mid \leq z_{1-\frac{\alpha}{2}}\right\},$$

esprimibile anche in forma esplicita

$$IC_{1-\alpha}(\phi) = \left\{\phi > 0 \colon \hat{\phi} \pm z_{1-\frac{\alpha}{2}} \sqrt{\widehat{Var}(\hat{\phi}))}\right\},$$

dove $z_{1-\frac{\alpha}{2}}$ è il quantile di livello $1-\alpha/2$ della distribuzione normale standard $N(0,1)$ e $\widehat{Var}(\cdot)$ è la varianza stimata.

La varianza è ottenuta dalla diagonale dell'inversa della matrice d'informazione osservata, questa è la matrice delle derivate parziali seconde della log-verosimiglianza cambiate di segno. Come già osservato nel paragrafo 1.3, l'indipendenza asintotica tra $\hat{\beta}$ e $\hat{\phi}$ fa sì che i blocchi non diagonali siano nulli. Di conseguenza

$$(j_{\phi\phi})^{-1} = j^{\phi\phi} = \frac{1}{j_{\phi\phi}}.$$

Nei modelli lineari generalizzati con legame canonico l'informazione osservata e il suo valore atteso, detto matrice d'informazione attesa, coincidono. Tuttavia, tale risultato non persiste per modelli come quello binomiale negativo dove sono simili con una numerosità sufficiente. Per il modello binomiale negativo tali quantità sono fornite da Lawless (1987).

Dato che il parametro di dispersione ϕ è limitato ai valori positivi, è possibile, prima del calcolo dell'intervallo di confidenza, utilizzare la trasformata logaritmica per riportare il suo supporto a $\mathbb{R}$, in tal caso la stima della varianza è ottenuta tramite il metodo delta (Salvan et al., 2020, Appendice D).

2.3 Intervalli di confidenza profilo

In modo equivalente all'intervallo di confidenza costruito basandosi sulla quantità di Wald, si può definire l'insieme di valori

$$\left\{\phi > 0 \colon \mid r_p(\phi) \mid \leq z_{1-\frac{\alpha}{2}}\right\},$$

come intervallo di confidenza profilo, dove

$$r_P(\phi) = \text{sgn}\,(\hat{\phi} - \phi)\ \sqrt{2(l_P(\hat{\phi}) - l_P(\phi))}.$$

Tale quantità approssimativamente pivotale è detta radice con segno del log-rapporto di verosimiglianza profilo, e nella sua definizione la funzione segno sgn$(\cdot)$, pari a 1 se l'argomento è positivo e -1 se negativo.

Gli intervalli di confidenza profilo sono solitamente preferibili rispetto a quelli di Wald, specialmente per campioni di ampiezza limitata, in quanto seguono la vera forma della funzione di verosimiglianza, anziché l'approssimazione parabolica come quelli basati sulla quantità di Wald che sono, dunque, centrati nella stima $\hat{\phi}$ e simmetrici. Inoltre, gli estremi dell'intervallo di confidenza profilo appartengono sempre allo spazio parametrico e godono, come lo stimatore di massima verosimiglianza, della proprietà di equivarianza rispetto a riparametrizzazioni.

Entrambi i metodi sono facilmente realizzabili in R. Infatti diversi pacchetti prevedono il calcolo di intervalli di confidenza. Il comando `confint`, presente in `stats`, permette la costruzione di intervalli di confidenza per oggetti di tipo `glm`, oltre che `lm`. Gli argomenti principali di tale funzione sono l'oggetto, il parametro su cui costruire l'intervallo e il livello di confidenza. Tale funzione costruisce intervalli di confidenza profilo, mentre per ottenere intervalli alla Wald è necessario utilizzare `confint.default`. Il pacchetto `VGAM` estende ulteriormente il comando agli oggetti `vglm`, la funzione `confintvglm` richiede gli stessi argomenti di `confint`, ma è necessario specificare il metodo con l'opzione `method`. Si noti che per parametri si intendono quelli dei coefficienti di regressione, β. Infine, l'intervallo di confidenza di Wald sul parametro τ del modello con risposta binomiale negativa è costruibile tramite la formula dato che la funzione `glm.nb` fornisce, oltre alla stima di $\kappa = 1/\tau$, `theta`, anche il relativo standard error `SE.theta`. Stessa considerazione vale per il modello binomiale negativo lineare in quanto `gamlss` fornisce le stime del parametro τ e del suo standard error, prestando attenzione al fatto che l'output è in scala lograitmica.

Capitolo 3

Intervalli di confidenza tramite bootstrap

3.1 Introduzione

Il bootstrap è una procedura di ricampionamento per permettere l'inferenza su un parametro di un modello. Il ricampionamento del campione osservato avviene seguendo diverse possibili assunzioni. In particolare, si distingue tra bootstrap parametrico, semi-parametrico e non parametrico. In questo capitolo si definiscono i residui calcolabili nei GLM, utilizzati nelle simulazioni semi-parametriche, si approfondiscono i diversi approcci per la definizione di procedure bootstrap e i diversi metodi per la costruzione di intervalli di confidenza.

3.2 Modelli lineari generalizzati e residui

Nel modello lineare normale i residui rappresentano la stima della componente d'errore e sono calcolati come la differenza tra una realizzazione della variabile risposta e la stima di questa. Nei modelli lineari generalizzati non è possibile definire in forma esplicita una componente d'errore e, dunque, i residui. Generalmente interessa studiare graficamente i residui, per verificare la correttezza della scelta di funzione legame, funzione di varianza e variabile esplicative inserite nel modello e la presenza di outliers. Per cui, nel definire i residui si desidera che, per un modello correttamente specificato, si distribuiscano approssimativamente come una normale standard. Tutte le verifiche sopra citate sono utili anche per la distinzione tra sovradispersione intrinseca e apparente, come spiegato nel paragrafo 1.2.

Dato y_i realizzazione della variabile risposta $Y_i,\ i = 1, \dots, n$, appartenente alla famiglia DE_1, il cui valore atteso μ_i è stimato da $\hat{\mu}_i$ e, di conseguenza, la cui funzione di varianza in $\hat{\mu}_i$ è $v(\hat{\mu}_i)$ e assunta la funzione $a_i(\phi) = \phi/\omega_i$ con ω_i pesi noti, in seguito si definiscono i principali tipi di residui (Salvan et al., 2020, paragrafo 2.4.2; Davison & Hinkley, 1997, paragrafo 7.2.2).

I residui della risposta

$$r_i^R = y_i - \hat{\mu}_i$$

seguono la forma di quelli definiti nel modello lineare normale. Un primo miglioramento può essere ottenuto standardizzando questi per la varianza della risposta, ottenendo così i residui di Pearson

$$r_i^P = \frac{y_i - \hat{\mu}_i}{\sqrt{\hat{Var}(Y_i)/\phi}} = \frac{y_i - \hat{\mu}_i}{\sqrt{v(\hat{\mu}_i)/\omega_i}}.$$

Viene usualmente applicata un'ulteriore standardizzazione per correggere l'effetto dell'utilizzo delle stime che provoca $Var(Y_i - \hat{\mu}_i) < Var(Y_i - \mu_i)$. I residui di Pearson standardizzati sono definiti come

$$r_i^{Ps} = \frac{y_i - \hat{\mu}_i}{\sqrt{\tilde{\phi}\,(1 - \hat{h}_{ii})\,v(\hat{\mu}_i)/\omega_i}}, \tag{3.1}$$

con $\tilde{\phi}$ stima calcolata con il metodo dei momenti e $\hat{h}_{ii}$ stima dell'i-esimo elemento sulla diagonale della matrice H generalizzata. La matrice H è misura dell'influenza del predittore lineare sul modello ed è definita come $H = W^{1/2}X(X^TWX)^{-1}X^TW^{1/2}$, dove W è una matrice diagonale con elementi $\omega_i = 1/((g'(\mu_i))^2\,Var(Y_i))$.

Seguendo un'idea simile a quella dei residui di Pearson standardizzati, è possibile definire i residui standardizzati del predittore lineare

$$r_i^{Ls} = \frac{g(y_i) - g(\hat{\mu}_i)}{\sqrt{\tilde{\phi}\,g'(\mu_i)^2(1 - \hat{h}_{ii})\,v(\hat{\mu}_i)/\omega_i}}. \tag{3.2}$$

Questi permettono di correggere un'eventuale asimmetria della distribuzione dei dati, che si riflette nei residui di Pearson, tuttavia la distribuzione approssimativamente normale standard è rispettata, con funzioni $g(\cdot)$ diverse dalla funzione identità, solo se ϕ è molto piccolo.

Un approccio alternativo è quello di replicare la proprietà dei modelli lineari generalizzati per cui la devianza equivale alla somma dei quadrati dei residui, così si ottengono i residui di devianza

$$r_i^D = \text{sgn}(y_i - \hat{\mu}_i)\,\sqrt{D_i}$$

e i residui di devianza standardizzati

$$r_i^{Ds} = \frac{\text{sgn}(y_i - \hat{\mu}_i)\sqrt{D_i}}{\sqrt{\tilde{\phi}\,(1 - \hat{h}_{ii})}}. \tag{3.3}$$

Con D_i si intende il contributo della singola osservazione alla devianza

$$D(y, \hat{\mu}) = 2\,\phi\,(l(y, \phi) - l(\hat{\mu}, \phi)),$$

misura della riduzione della bontà d'adattamento al passaggio dal modello saturo con $p = n$ a quello corrente con $p < n$ variabili esplicative. Nel caso in cui il modello in questione sia di Poisson la devianza risulta essere

$$D(y, \hat{\mu}) = 2\sum_{i=1}^{n} \big(y_i\ \log(y_i/\hat{\mu}_i) - y_i + \hat{\mu}_i\big),$$

mentre se si tratta di una binomiale negativa

$$D(y, \hat{\mu}) = 2\sum_{i=1}^{n} \big(y_i\ \log(y_i/\hat{\mu}) - (y_i + 1/\hat{\tau})\,(\log(1 + \hat{\tau} y_i) - \log(1 + \hat{\tau}\hat{\mu}_i))\big).$$

Si noti che la formulazione della devianza del modello per risposta binomiali negative tende a quello per il modello di Poisson per $\hat{\tau} \to 0^+$.

Inoltre, esistono i residui quantile che, per una variabile risposta con distribuzione discreta e funzione di ripartizione $F(y_i; \mu_i, \phi)$, sono pari a

$$r_i^Q = \Phi^{-1}(u_i),$$

dove $\Phi(\cdot)$ è la funzione di ripartizione della distribuzione normale standard e u_i è realizzazione di una variabile casuale uniforme nell'intervallo $(u_{i1}, u_{i2}]$ con $u_{i1} = \lim_{y_i \to y_i^-} F(y_i, \hat{\mu}_i, \tilde{\phi})$ e $u_{i2} = F(y_i, \hat{\mu}_i, \tilde{\phi})$. Tali residui mantengono casualità e distribuzione normale standard anche in casi di dispersione alta e distribuzioni discrete con media molto bassa. La loro definizione varia in caso di distribuzione continua della variabile risposta (Dunn & Smyth, 1996).

Infine, sono definiti i residui di Anscombe

$$r_i^A = \frac{A(y_i) - A(\hat{\mu}_i)}{A'(\hat{\mu}_i)\sqrt{v(\hat{\mu}_i)}},$$

dove la funzione $A(\cdot)$ è determinata con l'obiettivo che $A(Y_i),\ i = 1, \ldots, n$, abbia distribuzione più simile alla normale rispetto a $Y_i,\ i = 1, \ldots, n$, in modo tale da ridurre l'asimmetria dei residui calcolati su campioni di distribuzioni non normali. Per i modelli lineari generalizzati è definita come

$$A(\cdot) = \int_{y_i}^{\mu_i} \frac{1}{\sqrt[3]{v(\mu_i)}}\, d\mu_i,$$

dunque i residui di Anscombe hanno formulazione più complessa e comportamento simile a quello di r_i^{Ds}, ma tendono a essere normalizzati in modo tale che eterogeneità e outliers siano facilmente identificabili (Hilbe, 2011, paragrafo 5.1; McCullagh & Nelder, 1989, paragrafo 2.4.2).

3.3 Metodi basati sul bootstrap

A partire da un campione osservato si può distinguere tra simulazioni puramente non parametriche e simulazioni semi-parametriche. Il primo metodo consiste nel ricampionamento di casi e risulta essere inefficiente ma robusto rispetto all'eteroschedasticità, mentre il secondo prevede l'assunzione di corretta specificazione del modello generale, come di quasi-verosimiglianza o lineare generalizzato, per consentire il passaggio al ricampionamento dei residui.

Un'ulteriore alternativa sarebbero simulazioni parametriche, tuttavia esse si basano sul modello parametrico stimato per cui dati generati da modelli non correttamente specificati non avrebbero le stesse caratteristiche dei dati originali. In particolare, questo è il caso per dati di conteggio che presentano sovradispersione quando modellati tramite Poisson. Di conseguenza, risultano preferibili simulazioni non parametriche che permettono di generare dati senza la specificazione di un modello parametrico.

3.3.1 Bootstrap semi-parametrico

Gli algoritmi di ricampionamento semi-parametrico per la costruzione di intervalli di confidenza si basano sui residui. Data la non unicità nella definizione dei residui dei modelli lineari generalizzati, segue che nemmeno gli algoritmi di ricampionamento siano unici.

Si procede assumendo che i dati originali appartengano a una famiglia di dispersione esponenziale univariata per permettere maggiore variabilità delle osservazioni, inoltre si assume nuovamente $a_i(\phi) = \phi/\omega_i$.

Come descritto da Davison & Hinkley (1997, paragrafo 7.2.3) sono possibili 3 approcci al bootstrap semi-parametrico. Tutti seguono uno schema di R replicazioni di numerosità n, $i = 1, \ldots, n$. Per la costruzione di intervalli di confidenza con livello $1 - \alpha \geq 0.95$ risulta preferibile un valore di R maggiore di 999.

Il primo approccio prevede che le variabili risposta simulate y_i^* siano calcolate pari a

$$y_i^* = \hat{\mu}_i + \varepsilon_i^* \sqrt{\frac{\tilde{\phi}}{\omega_i} v(\hat{\mu}_i)}, \tag{3.4}$$

dove ε_i^*, $i = 1, \ldots, n$, è un campione casuale dei residui standardizzati di Pearson (3.1), corretti per la loro media $r_i^{Ps} - \bar{r}^{Ps}$. Si noti che non si assume un valore per ϕ sulla base del modello della famiglia DE_1 in cui i dati originali ricadono, facendo ciò si consente alla varianza di essere maggiore della media.

Il secondo approccio risulta essere una estensione del primo, nel caso in cui non sia possibile definire un legame esplicito tra y_i e ε_i. Si considera allora la trasformazione $g^{-1}(\cdot)$ dei valori simulati sulla scala del predittore lineare

$$y_i^* = g^{-1}\Big(\hat{\eta} + g'(\hat{\mu}_i)\, \varepsilon_i^* \sqrt{\tilde{\phi}\, v(\hat{\mu}_i)/\omega_i}\Big).$$

In questo caso, ε_i^*, $i = 1, \ldots, n$, è un campione dei residui r_i^{Ls} (3.2) e, a meno della funzione $g(\cdot)$ identità, non è necessario correggere per la media.

Infine, il terzo approccio prevede l'utilizzo dei residui di devianza standardizzati r_i^{Ds} (3.3), da cui ottenere un campione casuale di ε_i^*, $i = 1, \ldots, n$, e ricavare y_i^* da

$$\varepsilon_i^* = r_i^D(y_i^*, \hat{\mu}_i). \tag{3.5}$$

Nel caso in cui i residui risultino essere fortemente eterogenei, come risultato della presenza di sovradispersione nei dati, i metodi appena esposti potrebbero non tener conto interamente di tale sovradispersione. Ciò è verificabile tramite il grafico dei residui di Pearson standardizzati contro la distorsione stimata con la radice di $\hat{\mu}$. Se i residui mostrano andamento sistematico allora è possibile applicare gli algoritmi di ricampionamento dopo aver stratificato per gruppi di distorsione stimata simile.

Nonostante i valori osservati y_i siano interi e non negativi, tutti gli approcci non assicurano il rispetto di tali criteri dei dati generati. Una soluzione possibile è approssimare y_i^* al valore intero più vicino o a 0 se negativo. Tuttavia ciò può comportare distribuzioni distorte dei nuovi dati rispetto a quelli originali.

In conclusione, la costruzione degli intervalli di confidenza è possibile ottenendo le

stime $\hat{\beta}$ e $\tilde{\phi}$ da ciascun campione $y_1^*, \ldots, y_n^*$. Questo avviene per ogni replicazione, in tal modo si ha un campione $\tilde{\phi}_r,\ r = 1, \ldots, R$, da cui si ricava la distribuzione empirica di ϕ e dai quantili il rispettivo intervallo di confidenza.

3.3.2 Bootstrap non parametrico

L'approccio totalmente non parametrico prevede il ricampionamento dei casi. Per procedere è necessario distinguere il singolo caso, infatti, capita spesso che i dati di conteggio siano aggregati.

Una volta fatto ciò, si prosegue con il campionamento casuale con reinserimento dalle coppie di vettori $(x_i, y_i),\ i = 1, \ldots, n$, per andare a creare il nuovo campione $(x_1^*, y_1^*), \ldots, (x_n^*, y_n^*)$. Ciò è equivalente a campionare con reinserimento dagli indici delle unità $i = 1, \ldots, n$ e comprendere nel nuovo campione la coppia (x_i, y_i) corrispondente.

Dato che $y_i^*,\ i = 1, \ldots, n$, sono campionati casualmente, ogni campione simulato ha una certa informazione e campioni diversi porteranno a informazione diversa. Dunque l'inferenza dovrebbe essere specifica all'informazione dei dati, tuttavia tale variazione, per dataset di dimensione non troppo piccola, non è considerata significativa.

Dal campione simulato $y_1^*, \ldots, y_n^*$ nelle R replicazioni è possibile ricavare un intervallo di confidenza per il parametro ϕ in maniera equivalente a come descritto nella conclusione del paragrafo precedente. Tale metodo è detto basato sui percentili. Di seguito si illustrano metodi alternativi per la costruzione di intervalli di confidenza tramite bootstrap, facendo riferimento a Sartori & Guolo (2022).

3.4 Tipi di intervalli bootstrap

I metodi seguenti prevedono un campione bootstrap $y_i^*,\ i = 1, \ldots, n$, da cui ricavare i valori $\tilde{\phi}_r,\ r = 1, \ldots, R$. Per ottenere intervalli con livello di confidenza pari a $1 - \alpha$ sarà necessario tenere conto delle R replicazioni, in particolare il quantile di livello α sarà stimato dal $\alpha\,(R+1)$-esimo valore ordinato, dove R è tale da rendere $\alpha\,(R+1)$ un numero intero. Ad esempio, con $R = 999,\ R = 1999$.

Assumendo che lo stimatore sia approssimativamente normale $\tilde{\phi} \dot{\sim} N(\phi + \beta, \nu)$, è possibile costruire gli intervalli normali come

$$IC_{1-\alpha}(\phi) = \left(\tilde{\phi} - \beta - z_{1-\frac{\alpha}{2}}\sqrt{\nu},\ \tilde{\phi} - \beta + z_{\frac{\alpha}{2}}\sqrt{\nu}\right),$$

dove β e ν sono la distorsione e la varianza, stimabili dalle R replicazioni. Tali intervalli sono spesso non accurati in quanto prevedono un'assunzione forte sulla distribuzione

dello stimatore.

La costruzione degli intervalli basati sui percentili prevede l'utilizzo dei quantili empirici della distribuzione empirica ottenibile da $\tilde{\phi}_1, \ldots, \tilde{\phi}_R$, gli intervalli calcolati con tale metodo sono invarianti, al contrario di quelli normali, ma non molto accurati. L'intervallo di confidenza per ϕ risulterà essere

$$IC_{1-\alpha}(\phi) = \big(\tilde{\phi}_{(\frac{\alpha}{2}(R+1))},\ \tilde{\phi}_{((1-\frac{\alpha}{2})(R+1))}\big).$$

Un metodo alternativo, che segue un approccio altrettanto semplice, è quello degli intervalli studentizzati. Dato un campione $\tilde{\phi}_r,\ r = 1, \ldots, R$, si calcola la quantità di Wald

$$z_r^* = \frac{\tilde{\phi}_r - \tilde{\phi}}{\sqrt{\widehat{Var}_r(\tilde{\phi})}},\ r = 1, \ldots, R,$$

dove $\tilde{\phi}$ è la stima di ϕ basata sui dati $y_1, \ldots, y_n$ e e $\widehat{Var}(\tilde{\phi})$ una stima della varianza di $\tilde{\phi}_r$. Si utilizzano $z_1^*, \ldots, z_R^*$ per stimare la distribuzione di Z e i suoi quantili da cui l'intervallo di confidenza è calcolato come

$$IC_{1-\alpha}(\phi) = \Big(\tilde{\phi} - \big(\widehat{Var}_r(\tilde{\phi})^{1/2}\big)\, z^*_{((1-\frac{\alpha}{2})(R+1))},\ \tilde{\phi} - \big(\widehat{Var}_r(\tilde{\phi})^{1/2}\big)\, z^*_{(\frac{\alpha}{2}(R+1))}\Big).$$

Tali intervalli di confidenza risultano avere una maggiore accuratezza, tuttavia non sono invarianti e presuppongono la conoscenza della varianza di $\tilde{\phi}_r,\ r = 1, \ldots, R$. Per non incorrere in quest'ultima problematica è possibile applicare procedure come double bootstrap o jackknife, metodi che permettono la stima della varianza ma richiedono un numero di replicazioni elevato, oppure ottenere una approssimazione degli intervalli studentizzati tramite quelli bootstrap base.

Gli intervalli bootstrap base trattano la differenza $\tilde{\phi}_r - \tilde{\phi}$ come una quantità pivotale, permettendo di ottenere gli intervalli nonostante la varianza di ϕ non sia nota. L'intervallo diventa

$$IC_{1-\alpha}(\phi) = \big(2\,\tilde{\phi} - \tilde{\phi}_{((1-\frac{\alpha}{2})(R+1))},\ 2\,\tilde{\phi} - \tilde{\phi}_{(\frac{\alpha}{2}(R+1))}\big),$$

il quale rimane non invariante e risulta meno accurato di quello studentizzato.

Tramite dei miglioramenti degli intervalli basati sui percentili è possibile ottenere intervalli invarianti e accurati, gli estremi degli intervalli sono nuovamente i quantili della distribuzione empirica di ϕ

$$IC_{1-\alpha}(\phi) = \big(\tilde{\phi}_{(\alpha'(R+1))},\ \tilde{\phi}_{((1-\alpha'')(R+1))}\big),$$

dove α' e α'' sono opportunamente scelti per soddisfare le proprietà.

Nella categoria degli intervalli migliorati basati sui percentili ricadono gli intervalli corretti per la distorsione, detti intervalli BC. Si supponga che esista $h(\cdot)$ funzione monotona crescente tale che

$$h(\tilde{\phi}) \dot{\sim} N(h(\phi) - \omega, 1), \text{ da cui } h(\tilde{\phi}) - h(\phi) + \omega \dot{\sim} N(0,1).$$

Di conseguenza, i quantili utili alla costruzione degli intervalli BC sono

$$\alpha' = \Phi(2\hat{\omega} + z_{\alpha/2}) \text{ e } 1 - \alpha'' = \Phi(2\hat{\omega} + z_{1-\alpha/2})$$

dove $\Phi(\cdot)$ è la funzione di ripartizione di una distribuzione normale standard e $\hat{\omega}$ è ottenibile da $\Phi^{-1}(\cdot)$ della funzione di ripartizione della distribuzione bootstrap di ϕ calcolata in $\tilde{\phi}$.

Se la varianza di $h(\tilde{\phi})$ dipende da ϕ si utilizzano gli intervalli BC_a corretti per la distorsione e accelerati, in tal caso si ha $h(\cdot)$ monotona crescente tale che

$$h(\tilde{\phi}) \dot{\sim} N(h(\phi) - \omega\,\sigma_\phi, \sigma_\phi), \text{ da cui } \frac{h(\tilde{\phi}) - h(\phi)}{\sigma_\phi} + \omega \dot{\sim} N(0,1).$$

Inoltre, si assume che $\sigma_\phi = \sigma_{\phi_0}\,(1 + a\,(h(\phi) - h(\phi_0)))$ con ϕ_0 valore di ϕ per cui $\sigma_{\phi_0} = 1$. I quantili degli estremi dell'intervallo sono quelli corrispondenti a

$$\alpha' = \Phi\Big(\omega + \frac{\omega + z_{\alpha/2}}{1 - a\,(\omega + z_{\alpha/2})}\Big) \text{ e } 1 - \alpha'' = \Phi\Big(\omega + \frac{\omega + z_{1-\alpha/2}}{1 - a\,(\omega + z_{1-\alpha/2})}\Big)\,,$$

dove ω e a sono stimati da $\hat{\omega}$ e $\hat{a}$ (Sartori & Guolo, 2022, slide 333). Gli intervalli di confidenza BC_a sono invarianti e, secondo la teoria, hanno accuratezza pari a quella degli intervalli studentizzati. Si nota che se negli intervalli BC ω è pari a 0, allora la distorsione in mediana di $h(\tilde{\phi})$ è nulla e il metodo è equivalente a quello basato sui percentili. Allo stesso modo, se negli intervalli BC_a l'accelerazione di σ_ϕ è nulla, allora il metodo corrisponde a quello BC.

Nel capitolo successivo si presenterà l'applicazione a due casi di studio di tutti i metodi appena descritti.

Capitolo 4

Esempi di applicazione

4.1 Introduzione

Nei prossimi paragrafi si presenterà l'applicazione in R dei metodi descritti nei capitoli precedenti su *dataset* presenti nel pacchetto `boot` (A. Canty & B. D. Ripley, 2024), utile per la costruzioni di procedure bootstrap. Per ciascun *dataset* si considerano bootstrap parametrico basato sul modello di Poisson, bootstrap semi-parametrico con l'utilizzo di residui di Pearson e quelli di devianza, e bootstrap totalmente non parametrico. Per ognuno di questi sono costruiti intervalli di confidenza per il parametro ϕ attraverso i diversi metodi. Sono riportati per il confronto gli intervalli di confidenza normali, basati sui percentili, base, studentizzati, BC e BC_a. L'implementazione in R dei metodi è fornita in Appendice. Si noti che nel modello Poisson ϕ è fissato pari a 1. In tale caso, nel bootstrap parametrico si ricampiona una statistica corrispondente allo stimatore basato sul metodo dei momenti.

In seguito si pone il livello di confidenza degli intervalli di confidenza pari a $1 - \alpha = 0.95$ e vengono svolte $R = 1999$ replicazioni, si tratta di un valore basso per l'affidabilità dei metodi. Tuttavia, per rendere possibile la stima della varianza e, dunque, il calcolo degli intervalli studentizzati è necessario ricorrere al metodo del double bootstrap con un'ulteriore bootstrap di $P = 999$ simulazioni, ciò porta ad un aumento delle replicazioni totali.

4.2 Esempio 1: *cloth*

Il *dataset cloth* (Bissell, 1972) contiene informazioni riguardanti il numero di difetti riscontrati in 32 rotoli di tessuto (y), in relazione alla loro lunghezza (x). La presenza

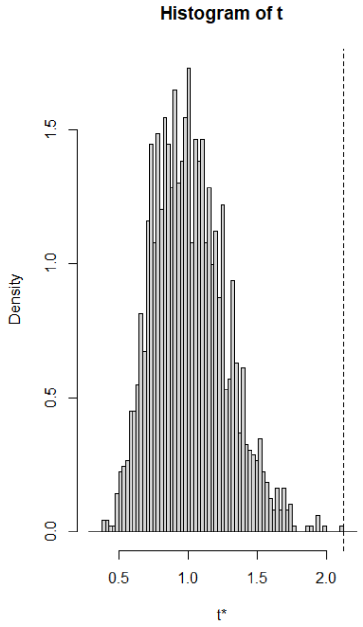
di sovradispersione, a seguito dell'adattamento dei dati ad un modello Poisson, è confermata dalla statistica di Pearson e la stima di ϕ basata sul metodo dei momenti è $\tilde{\phi} = 2.122$.

I risultati delle procedure bootstrap sono rappresentati nella Figura 4.1. Per ogni algoritmo di ricampionamento presentato nel paragrafo 3.3 vengono rappresentate le distribuzioni delle replicazioni nella stima del parametro di sovradispersione, denominato t negli istogrammi. La linea verticale tratteggiata rappresenta il valore della stima osservato nei dati.

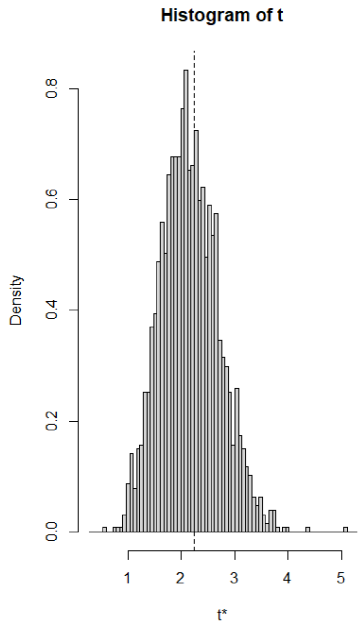
Si noti che la distribuzione ottenuta tramite bootstrap parametrico ricade interamente su valori inferiori alla stima di ϕ e perciò non sarà possibile calcolare gli intervalli di confidenza BC e BC_a. I bootstrap semi-parametrici producono distribuzioni simili a quelli non parametrici, tuttavia è presente una leggera distorsione della linea verticale rispetto al valore di $\tilde{\phi}$, essa è causata dalla limitazione a valori interi e non negativi delle variabili risposta simulate. Infine, oltre al bootstrap non parametrico, viene rappresentata la distribuzione del bootstrap non parametrico stratificato per i quartili della variabile x. Rispetto ai metodi semi-parametrici, quelli non parametrici presentano una dispersione inferiore attorno a $\tilde{\phi}$.

A partire da un oggetto di tipo `boot`, è possibile ottenere gli intervalli di confidenza tramite la funzione `boot.ci`. L'intervallo di confidenza BC è l'unico non implementato, è presente la sua generalizzazione BC_a che non assume accelerazione nulla, dunque per il suo calcolo è stata codificata la funzione `ic.BC`, presente in Appendice.

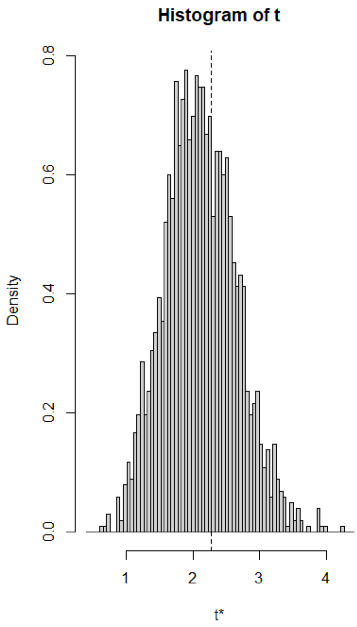
Nella Tabella 4.1 sono raccolti gli intervalli di confidenza che risultano calcolabili per tutti i metodi bootstrap. Come già osservato, gli intervalli di confidenza calcolati sulla base del bootstrap parametrico non sono affidabili, non contenendo $\tilde{\phi}$ e portando a conclusioni discordanti tra loro. I restanti intervalli di confidenza hanno estremi di valori simili tra loro, tuttavia si tende a preferire gli intervalli di confidenza studentizzati che hanno maggiore accuratezza. Infatti, gli intervalli normali assumono che la distribuzione del bootstrap sia normale, quelli percentili sono poco affidabili, come osservabile da quello calcolato con simulazione non parametrica, e quelli base sono una semplificazione di quelli studentizzati.



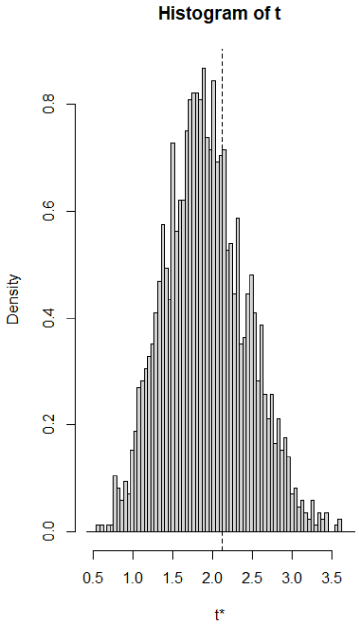
(a) Parametrico



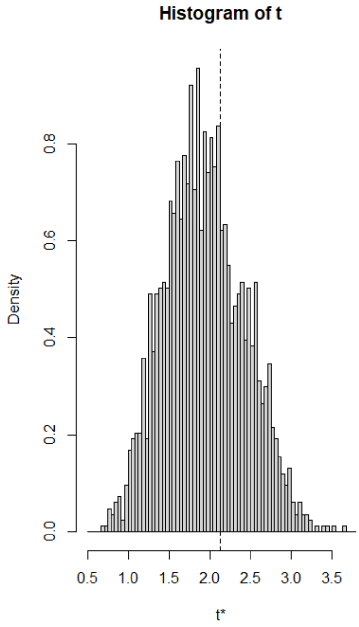
(b) Semi-parametrico r^P



(c) Semi-parametrico r^D



(d) Non parametrico



(e) Non parametrico stratificato

FIGURA 4.1: Distribuzioni bootstrap per $\tilde{\phi}$ in *cloth*

TABELLA 4.1: Intervalli di confidenza normali, percentili, base e studentizzati per ϕ tramite bootstrap con diversi metodi di ricampionamento nel *dataset cloth*.

	Metodo di costruzione			
Schema di campionamento	Normale	Percentile	Base	Studentizzato
Parametrico	(2.724, 3.744)	(0.574, 1.573)	(2.671, 3.670)	(2.886, 8.735)
Semi-parametrico da r^P	(1.257, 3.393)	(1.179, 3.282)	(1.206, 3.310)	(1.203, 3.328)
Semi-parametrico da r^D	(1.401, 3.486)	(1.118, 3.226)	(1.330, 3.438)	(1.273, 3.458)
Non parametrico	(1.329, 3.325)	(1.006, 2.960)	(1.284, 3.238)	(1.283, 3.250)
Non parametrico stratificato	(1.360, 3.265)	(1.057, 2.898)	(1.346, 3.186)	(1.330, 3.217)

Nella Tabella 4.2 sono presentati gli intervalli di confidenza di tipo BC e BC_a. Questi, in base alla teoria richiamata nel paragrafo 3.4, raggiungono la stessa accuratezza degli intervalli studentizzati, ma con migliori proprietà teoriche quali l'invarianza rispetto trasformazioni. Dalla Figura 4.2 si osserva che gli intervalli BC_a hanno estremi in valore superiore sia a quelli BC che a quelli studentizzati e ampiezza maggiore. Viceversa, gli intervalli BC e studentizzati risultano essere i più simili tra loro.

TABELLA 4.2: Intervalli di confidenza BC e BC_a per ϕ tramite bootstrap con diversi metodi di ricampionamento nel *dataset cloth*.

	Metodo di costruzione	
Schema di campionamento	BC	BC_a
Semi-parametrico da r^P	(1.173, 3.274)	(1.431, 3.723)
Semi-parametrico da r^D	(1.188, 3.268)	(1.505, 3.920)
Non parametrico	(1.368, 3.420)	(1.417, 3.598)
Non parametrico stratificato	(1.389, 3.272)	(1.442, 3.556)

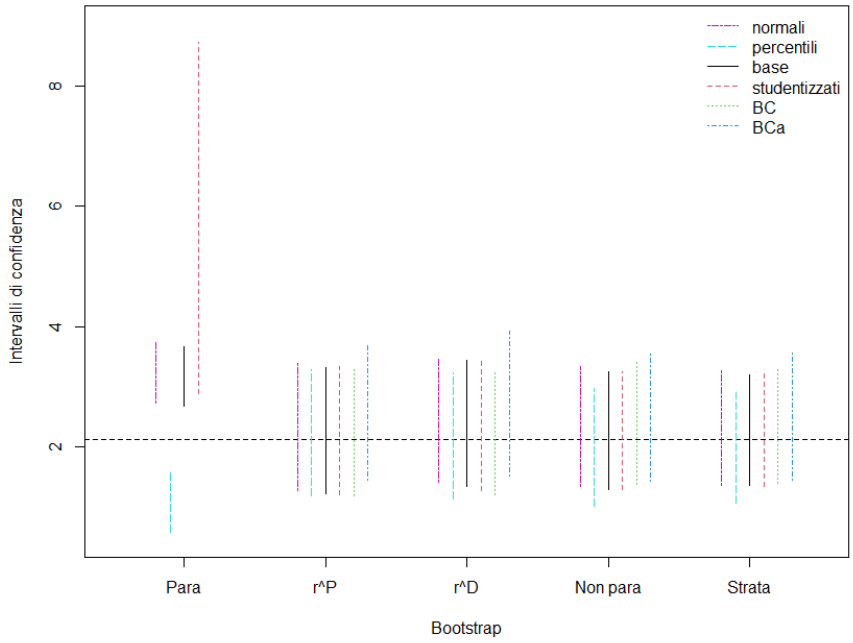
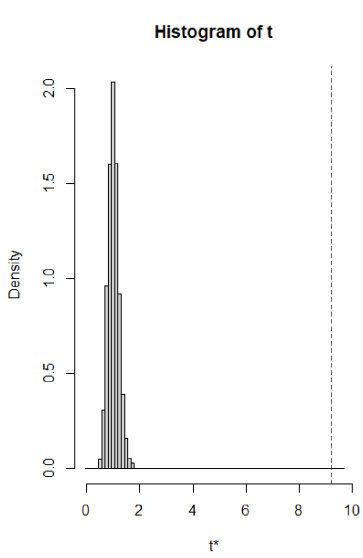


FIGURA 4.2: Intervalli di confidenza per ϕ tramite bootstrap nel *dataset cloth*

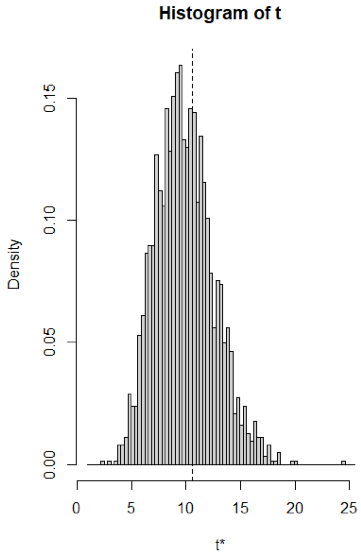
4.3 Esempio 2: *bigcity*

Il *dataset bigcity* (Cochran, 1977) contiene i dati della numerosità in migliaia della popolazione di 49 città americane dove x è la variabile risposta che indica la popolazione nel 1930 e u è la variabile esplicativa che quantifica la numerosità nel 1920. La stima di ϕ basata sul metodo dei momenti è pari a 9.21. Al campione vengono applicati i 5 metodi bootstrap utilizzati anche nel paragrafo precedente, i risultati sono rappresentati nella Figura 4.3 dove t rappresenta la stima del parametro per cui si ricampiona e la linea orizzontale tratteggiata la stima di questo nei dati.

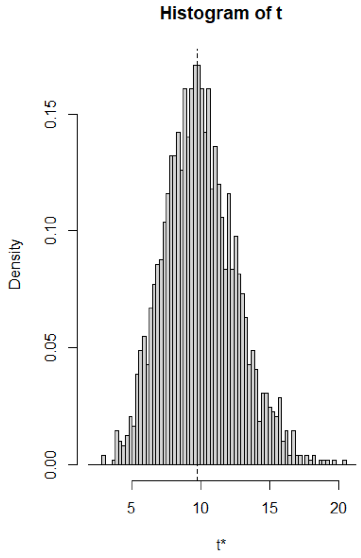
Si nota nuovamente che la stima di ϕ non rientra nella distribuzione dei valori ottenuti tramite simulazioni parametriche, ciò è ancora più evidente rispetto al *dataset cloth* in quanto i dati hanno una dispersione superiore. Le distribuzioni risultanti dalle procedure bootstrap semi-parametriche presentano entrambe distorsione rispetto a $\tilde{\phi}$ e leggera asimmetria a destra, più forti per la procedura basata sui residui di Pearson dove la linea verticale è $t = 10.56$. Le procedura bootstrap non parametriche danno risultati simili, quella stratificata ha minore dispersione con frequenze più elevate attorno i valori centrali.



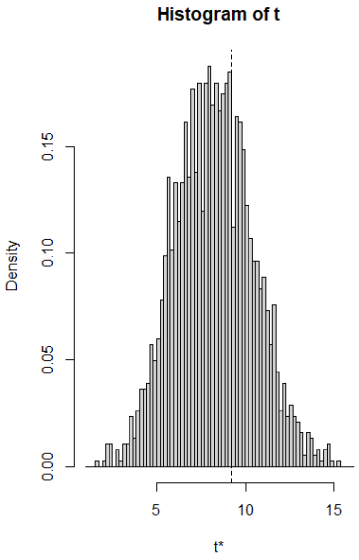
(a) Parametrico



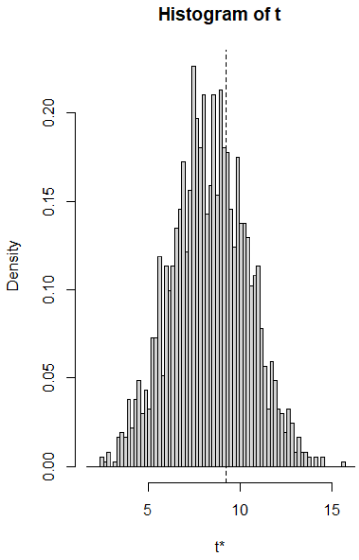
(b) Semi-parametrico r^P



(c) Semi-parametrico r^D



(d) Non parametrico



(e) Non parametrico stratificato

FIGURA 4.3: Distribuzioni bootstrap per $\tilde{\phi}$ in *bigcity*

Gli intervalli di confidenza basati sui metodi bootstrap sono contenuti nelle Tabelle 4.3-4.4. Si nota che quelli costruiti sulla distribuzione ottenuta da bootstrap parametrico non sono affidabili, in particolare l'intervallo studentizzato in quanto per il suo calcolo la varianza dello stimatore è ottenuta tramite un boostrap non parametrico. Per ciò che riguarda le procedure semi-parametriche gli intervalli hanno ampiezza maggiore rispetto quelle non parametriche. Gli intervalli basati sul ricampionamento di r^P sono meno centrati in $\tilde{\phi}$ e l'intervallo ottenuto tramite il metodo BC_a è di estremi in valore superiore agli altri. Le due procedure non parametriche producono intervalli di confidenza basati su tutti i metodi di costruzione molto simili tra loro. Una rappresentazione grafica è la Figura 4.4, la quale non include l'intervallo di confidenza studentizzato basato su bootstrap parametrico.

TABELLA 4.3: Intervalli di confidenza normali, percentili, base e studentizzati per ϕ tramite bootstrap con diversi metodi di ricampionamento nel *dataset bigcity*.

	Metodo di costruzione			
Schema di campionamento	Normale	Percentile	Base	Studentizzato
Parametrico	(17.014, 17.813)	(0.633, 1.444)	(16.975, 17.786)	(67.808, 187.030)
Semi-parametrico da r^P	(5.970, 16.510)	(5.190, 15.730)	(5.390, 15.930)	(5.360, 15.970)
Semi-parametrico da r^D	(4.365, 14.513)	(5.336, 15.535)	(3.819, 14.019)	(3.668, 14.141)
Non parametrico	(5.867, 14.520)	(4.132, 12.678)	(5.741, 14.286)	(5.889, 14.044)
Non parametrico stratificato	(6.045, 14.138)	(4.203, 12.367)	(6.051, 14.216)	(6.047, 14.237)

TABELLA 4.4: Intervalli di confidenza BC e BC_a per ϕ tramite bootstrap con diversi metodi di ricampionamento nel *dataset bigcity*.

	Metodo di costruzione	
Schema di campionamento	BC	BC_a
Semi-parametrico da r^P	(4.629, 14.253)	(6.800, 19.990)
Semi-parametrico da r^D	(4.296, 13.943)	(5.619, 16.124)
Non parametrico	(5.878, 14.663)	(6.141, 15.292)
Non parametrico stratificato	(5.990, 13.722)	(6.189, 14.568)

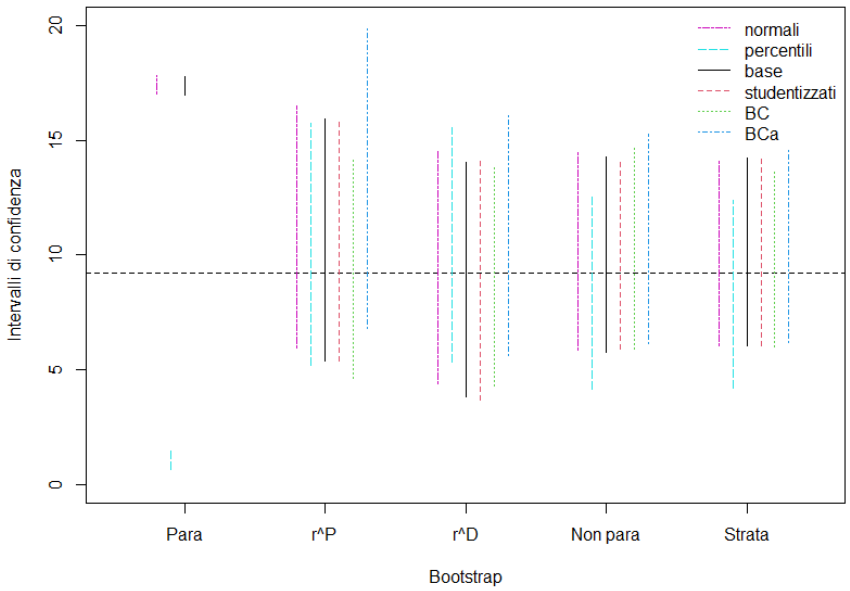


FIGURA 4.4: Intervalli di confidenza per ϕ tramite bootstrap nel *dataset bigcity*

Nel capitolo successivo si valuteranno tramite simulazione le coperture degli intervalli di confidenza con i diversi algoritmi di campionamento e metodi per la costruzione degli intervalli, al variare del valore del parametro di sovradispersione.

Capitolo 5

Simulazioni

5.1 Coperture degli intervalli di confidenza

In questo capitolo si valuteranno le copertura empiriche degli intervalli di confidenza per il parametro di sovradispersione, la cui copertura teorica è fissata a 0.95. Le simulazioni vengono effettuate per diversi valori di ϕ del modello di quasi-verosimiglianza che soddisfa le ipotesi del secondo ordine. Si valutano prima le coperture per $\phi = 1$, generando i campioni da una variabile casuale Poisson, e successivamente per $\phi > 1$, generando i campioni dalla distribuzione binomiale negativa lineare. Le conclusioni vengono raccolte per le diverse procedure bootstrap e tipi di intervalli di confidenza su $N = 1000$ campioni simulati di dimensione n.

5.2 Simulazioni con $\phi = 1$

Per valutare le coperture empiriche su campioni con parametro di sovradispersione pari a 1 i campioni sono generati da una distribuzione Poisson. Questa ha media $\mu_i = g(\eta_i)^{-1}$ e predittore lineare $\eta_i = \boldsymbol{x}_i\ \beta = \beta_1 + \beta_2\, x_{i2}$ con funzione di legame $g(\cdot) = \log(\cdot)$. Dove $\beta = (1.1, 0.7)$ e x_{i2} uguale al vettore dei valori x nel *dataset cloth*, per cui $n = 32$. La codifica della funzione per generare i campioni è presente in Appendice.

Nella Tabella 5.1 sono raccolte le coperture empiriche per ogni combinazione di metodo bootstrap e tipo d'intervallo di confidenza. Si utilizzano le procedure di bootstrap semi-parametrico basato su residui di Pearson e di devianza, e non parametrico, riducendo R a 999 e mantenendo $P = 999$. La simulazione del bootstrap parametrico viene omessa in quanto, anche dagli esempi forniti nel Capitolo 4, si osserva che il modello parametrico assunto ha ϕ fissato pari a 1.

Si osserva che nel caso non parametrico risulta preferibile l'utilizzo degli intervalli BC_a, mentre essi presentano un peggioramento rispetto a quelli BC negli schemi semi-parametrici. Inoltre, gli intervalli base e studentizzati portano a risultati equivalenti per ognuno degli approcci al ricampionamento considerati. Infine, gli intervalli di confidenza normali sono, per tutte le procedure bootstrap, quelli con copertura empirica inferiore.

TABELLA 5.1: Copertura empirica di intervalli di confidenza per dati simulati da Poisson ($\phi = 1$)

	Metodo di costruzione IC					
Schema di campionamento	Normale	Percentile	Base	Studentizzato	BC	BC_a
Semi-parametrico da r^P	0.868	0.906	0.900	0.899	0.906	0.892
Semi-parametrico da r^D	0.893	0.909	0.909	0.909	0.909	0.906
Non parametrico	0.890	0.879	0.889	0.889	0.897	0.913

5.3 Simulazioni con $\phi > 1$

Per permettere il confronto dei risultati all'aumentare della sovradispersione i campioni sono simulati da una distribuzione binomiale negativa lineare, seguendo la notazione introdotta nel paragrafo 1.4, con $\mu_i = \exp(\beta_1 + \beta_2 x_{i2})$ e rapporto media - varianza $(1 + \tau)$. Sono fissati i parametri $\beta = (1.1, 0.7)$ e $\tau = 2.5$, in modo tale che $\phi = 1 + \tau = 3.5$. Nuovamente si procede ricavando i valori x dal *dataset cloth*, la codifica della funzione che permette le simulazioni è fornita in Appendice.

La Tabella 5.2 contiene le coperture empiriche degli intervalli di confidenza ottenuti tramite metodi bootstrap parametrici e non parametrici con $R = 999$. Per ottenere l'intervallo di confidenza BC_a basato sul ricampionamento di tipo parametrico si utilizza la forma approssimata con i valori jackknife, che causa la perdita della proprietà d'invarianza, in quanto non è possibile il calcolo delle componenti individuali di influenza per tale bootstrap.

Si osserva che, sia nel caso parametrico che in quello non parametrico, gli intervalli con copertura empirica più vicina a quella teorica sono quelli normali e BC_a. Al contrario, quelli con copertura più bassa sono quelli studentizzati, risultato in contrasto con le proprietà teoriche di tali intervalli. Inoltre, gli intervalli di confidenza percentili costruiti sulla base del bootstrap parametrico sono soddisfacenti, ciò è provocato dalla corretta specificazione del modello.

Infine, si noti che gli intervalli considerati sono costruiti per il parametro τ, quelli del parametro ϕ sono ottenuti a seguito di una traslazione.

TABELLA 5.2: Copertura empirica di intervalli di confidenza per dati simulati da NB1 ($\phi > 1$)

	Metodo di costruzione IC					
Schema di campionamento	Normale	Percentile	Base	Studentizzato	BC	BC_a
Parametrico	0.972	0.931	0.90	0.709	0.932	0.932
Non parametrico	0.941	0.886	0.915	0.728	0.887	0.952

Dal confronto tra le Tabelle 5.1-5.2 è possibile notare che le conclusioni sono discordanti, in particolare rimane dubbia la correttezza del metodo normale, ritenuto affidabile solo tramite le simulazioni da distribuzione binomiale negativa lineare. Gli intervalli studentizzati non raggiungono l'accuratezza teorica del metodo. In particolare nel caso $\phi > 1$, dove la copertura empirica è inferiore a 0.75 in entrambi gli schemi di ricampionamento. Tuttavia, ciò può essere giustificato dal valore di R utilizzato per le simulazioni da NB1, dove il numero di replicazioni totali è di molto inferiore a quello del paragrafo precedente ($R(P + 1)$) per cui l'accuratezza della copertura potrebbe averne risentito. In ogni caso si tratta di un aspetto che richiede ulteriori approfondimenti. Il metodo che porta alla copertura migliore in entrambi i casi studiati è quello BC_a basato sul boostrap non parametrico.

Conclusioni

La sovradispersione nei dati di conteggio è un fenomeno che, se non trattato correttamente, porta a conclusioni fuorvianti. L'adattamento dei dati con modelli di variabilità più flessibile permettono anche la stima della misura della sovradispersione presente nei dati: il parametro di sovradispersione, esso e il suo intervallo di confidenza risultano essere d'interesse. D'interesse ulteriore potrebbe essere la costruzione di intervalli di confidenza per la trasformata logaritmica del parametro di sovradispersione, ottenibile solamente per i metodi di costruzione non invarianti, in modo tale che il supporto non sia limitato a valori positivi.

La costruzione dell'intervallo di confidenza può basarsi su schemi di ricampionamento parametrico o non. In entrambi i casi sono possibili diversi metodi di costruzione basati su procedure bootstrap. In aggiunta, nel caso in cui il modello utilizzato ricada nei modelli lineari generalizzati è possibile utilizzare un approccio semi-parametrico basato sui residui.

Sulla base di 1000 simulazioni da due distribuzioni diverse, per diversi valori del parametro di sovradispersione, si conclude che l'approccio parametrico, considerato per $\phi > 1$, porta a coperture soddisfacenti ad eccezione di quelli base e studentizzato. Gli intervalli di confidenza basati su procedure semi-parametriche, valutate per $\phi = 1$, producono coperture empiriche simili al variare dei metodi di costruzione e migliori per il ricampionamento basato su residui di devianza. L'approccio non parametrico porta a conclusioni coerenti rispetto le simulazioni, risulta preferibile la costruzione d'intervallo tramite metodo BC_a. Invece, per ciò che riguarda i metodi di costruzione gli intervalli di confidenza che portano a conclusioni migliori al variare del valore di ϕ e di schema di ricampionamento sono BC e BC_a, seguiti da quelli base.

Appendice

Applicazione a *dataset*: esempio *cloth*

In questo paragrafo si presenta l'applicazione al *dataset cloth*, la codifica per l'esempio *bigcity* risulta essere equivalente con variabile risposta x e variabile esplicativa u.

Caricamento pacchetti e stime iniziali

Si procede con il caricamento dei pacchetti utili allo svolgimento delle applicazioni, l'adattamento del modello di Poisson e la stima del parametro di sovradispersione tramite il metodo dei momenti.

```
library(boot)

cloth.po <- glm(y~x, poisson, data=cloth)
summary(cloth.po)

X2 <- sum(residuals(cloth.po, type='pearson')^2)
pchisq(X2, cloth.po$df.residual, lower.tail=F)
X2/cloth.po$df.residual
```

Inoltre, si ottiene la media dei valori di y che sarà utilizzata in diversi schemi di ricampionamento.

```
mu.y <- mean(cloth$y)
```

Bootstrap parametrico

La funzione `boot` dell'omonimo pacchetto permette di ottenere bootstrap non parametrici e parametrici. Per questi ultimi è necessario definire due funzioni: la prima per il calcolo della stima del parametro d'interesse, in questo caso $\tilde{\phi}$, mentre la seconda per simulare valori con proprietà analoghe al campione. All'interno della prima funzione è

presente il bootstrap non parametrico annidato per il calcolo della stima della varianza.

```
phi.po <- function(data){
  glm.po <- glm(data$y~data$x, poisson)
  var.phi <- function(data,i){ # bootstrap annidato per il calcolo
                               # della varianza
    d <- data[i,]
    glm.po <- glm(d$y~d$x, poisson)
    sum(residuals(glm.po, type='pearson')^2)/glm.po$df.residual
  }
  boot.np.var <- boot(data, statistic = var.phi, R=999)
  cbind(sum(residuals(glm.po, type='pearson')^2)/glm.po$df.residual,
   var(boot.np.var$t)*(nrow(boot.np.var$t)-1)/nrow(boot.np.var$t))
}
phi.sim <- function(data, mle){
  d <- data
  d$y <- rpois(n=nrow(data), mle)
  d
}
phi.para <- boot(cloth, statistic = phi.po, R=1999, sim='parametric',
   ran.gen = phi.sim, mle=mu.y)
```

Bootstrap semi-parametrico

Le procedure di bootstrap semi-parametrico utilizzano l'opzione della funzione `boot` per il ricampionamento non parametrico che richiede solo una funzione per il calcolo del parametro da simulare. I residui di Pearson e di devianza sono ottenuti tramite la funzione `glm.diag` del pacchetto `boot` che permette il calcolo di residui e statistiche utili a partire da un oggetto modellato con `glm`.

Per il ricampionamento basato sui residui di Pearson si implementa la formula (3.4), dopo aver corretto i residui per la loro media.

```
cloth.diag <- glm.diag(cloth.po)
cloth.rp <- data.frame(cloth, fit=fitted(cloth.po),
                       pea=(cloth.diag$rp-mean(cloth.diag$rp)))
boot.fun.rp <- function(data, i){
  y.fun <- data$fit+sqrt(data$fit)*data$pea[i]
  y.fun <- round(y.fun)
  y.fun[y.fun<0] <- 0
  d.po <- glm(y.fun~data$x, poisson)
  var.phi <- function(data,j){
```

```
    d <- data[j,]
    glm.po <- glm(d$y~d$x, poisson)
    sum(residuals(glm.po, type='pearson')^2)/glm.po$df.residual
  }
  boot.np.var <- boot(data, statistic = var.phi, R=999)
  cbind(sum(residuals(d.po, type='pearson')^2)/d.po$df.residual,var(
   boot.np.var$t)*(nrow(boot.np.var$t)-1)/nrow(boot.np.var$t))
}
phi.boot.rp <- boot(cloth.rp, boot.fun.rp, R=1999)
```

Allo stesso modo, si codifica la procedura per il metodo bootstrap basato sui residui di devianza che richiede l'inversione della relazione (3.5). Per far ciò, si simulano possibili valori della variabile risposta e si calcolano i corrispondenti valori dei residui di devianza. Tramite la funzione `smooth.spline` del pacchetto `stats` è possibile valutare la relazione tra y e r^D e ottenere i valori di y corrispondenti ai residui di devianza standardizzati del *dataset*.

```
y.sim <- seq(from=min(cloth$y), to=max(cloth$y),
              length.out=9999) # y simulate
rd.sim <- sign(y.sim-mu.y)*sqrt(2*(y.sim*log(y.sim/mu.y)
          -y.sim+mu.y)) # r^D corrispondenti
sim <- smooth.spline(rd.sim,y.sim) # si valuta la relazione tra y e rd
z <- predict(sim, cloth.diag$rd)$y # r^Ds su cui valutare y
cloth.rd <- data.frame(cloth, z)
boot.fun.rd <- function(data, i){
  y.fun <- data$z[i]
  y.fun <- round(y.fun)
  y.fun[y.fun<0] <- 0
  d.po <- glm(y.fun~data$x, poisson)
  var.phi <- function(data,j){
    d <- data[j,]
    glm.po <- glm(d$y~d$x, poisson)
    sum(residuals(glm.po, type='pearson')^2)/glm.po$df.residual
  }
  boot.np.var <- boot(data, statistic = var.phi, R=999)
  cbind(sum(residuals(d.po, type='pearson')^2)/d.po$df.residual,var(
   boot.np.var$t)*(nrow(boot.np.var$t)-1)/nrow(boot.np.var$t))
}
phi.boot.rd <- boot(cloth.rd, boot.fun.rd, R=1999)
```

Bootstrap non parametrico

L'implementazione di un bootstrap non parametrico prevede la sola definizione della funzione per il calcolo della statistica d'interesse.

```
phi.np <- function(data,i){
  d <- data[i,]
  glm.po <- glm(d$y~d$x, poisson)
  var.phi <- function(data,i){
    d <- data[i,]
    glm.po <- glm(d$y~d$x, poisson)
    sum(residuals(glm.po, type='pearson')^2)/glm.po$df.residual
  }
  boot.np.var <- boot(data, statistic = var.phi, R=999)
  cbind(sum(residuals(glm.po, type='pearson')^2)/glm.po$df.residual,
   var(boot.np.var$t)*(nrow(boot.np.var$t)-1)/nrow(boot.np.var$t))
}
phi.boot.np <- boot(cloth, statistic = phi.np, R=1999)
```

La funzione `boot` permette anche la stratificazione tramite l'opzione `strata`. Si definiscono gli strati come i quantili della variabile x.

```
s <-cut(cloth$x,quantile(cloth$x,0:4/4),include.lowest = T)
phi.boot.np.s <- boot(cloth, statistic = phi.np, R=1999, strata=s)
```

Funzione `ic.BC`

Gli intervalli di confidenza previsti dalla funzione `boot.ci` non comprendono gli intervalli di confidenza BC, perciò di definisce una funzione specifica al loro calcolo che segua la definizione fornita nel paragrafo 3.4 e prenda come input un oggetto di tipo `boot`.

```
ic.BC <- function(bootт){
  w <- qnorm(mean(bootт$t[,1] <= X2/cloth.po$df.residual))
  z <- qnorm(c(.025,.975))
  q_bc <- pnorm(2*w+z)
  quantile(bootт$t[,1], q_bc)
}
```

Risultati

Per ottenere i grafici della Figura 4.1 si utilizza la funzione `plot` di un oggetto di tipo `boot`.

```
plot(phi.para)
plot(phi.boot.rp)
plot(phi.boot.rd)
plot(phi.boot.np)
plot(phi.boot.np.s)
```

Per gli intervalli di confidenza si utilizzano le funzioni `boot.ci` e `ic.BC`. La prima, oltre all'oggetto a cui applicare i metodi, richiede anche il tipo di intervallo di confidenza desiderato `type` con default `'all'`. Dunque, per non incorrere in un errore, è necessario specificare i soli metodi calcolabili tramite bootstrap parametrico.

```
boot.ci(boot.out = phi.para, type=c('norm','perc','basic','stud'))
boot.ci(boot.out = phi.boot.rp)
ic.BC(phi.boot.rp)
boot.ci(boot.out = phi.boot.rd)
ic.BC(phi.boot.rd)
boot.ci(boot.out = phi.boot.np)
ic.BC(phi.boot.np)
boot.ci(boot.out = phi.boot.np.s)
ic.BC(phi.boot.np.s)
```

Simulazioni: $\phi = 1$

Funzioni bootstrap

Vengono utilizzate funzioni simili a quelle definite in precedenza per impostare bootstrap parametrico, semi-parametrico e non parametrico. Tuttavia, per permettere un calcolo più veloce, si definisce una nuova funzione `var.np` per la stima della varianza dello stimatore tramite bootstrap annidato. Dato che ciò avviene esternamente alle funzioni `boot.fun.rp`, `boot.fun.rd` e `phi.np` la nuova codifica è la seguente. Si noti che l'output di tali funzioni continua a essere composto tramite `cbind`, il secondo elemento verrà sovrascritto dalle stime della varianza.

```
# bootstrap non parametrico annidato per calcolo varianza
var.np <- function(data,i){
  var.phi <- function(data,i){
    d <- data[i,]
    glm.po <- glm(d$y~d$x, poisson)
    sum(residuals(glm.po, type='pearson')^2)/glm.po$df.residual
  }
```

```
  boot.np.var <- boot(data, statistic = var.phi, R=999)
  cbind(var(boot.np.var$t)*(nrow(boot.np.var$t)-1)/nrow(boot.np.var$t)
   )
}

# bootstrap semi-parametrico con r_p
boot.fun.rp <- function(data, i){
  y.fun <- data$fit+sqrt(data$fit)*data$pea[i]
  y.fun <- round(y.fun)
  y.fun[y.fun<0] <- 0
  d.po <- glm(y.fun~data$x, poisson)
  cbind(sum(residuals(d.po, type='pearson')^2)/d.po$df.residual,1)
}

# bootstrap semi-parametrico con r_d
boot.fun.rd <- function(data, i){
  y.fun <- data$z[i]
  y.fun <- round(y.fun)
  y.fun[y.fun<0] <- 0
  d.po <- glm(y.fun~data$x, poisson)
  cbind(sum(residuals(d.po, type='pearson')^2)/d.po$df.residual,1)
}

# bootstrap non parametrico
phi.np <- function(data,i){
  d <- data[i,]
  glm.po <- glm(d$y~d$x, poisson)
  cbind(sum(residuals(glm.po, type='pearson')^2)/glm.po$df.residual,1)
}
```

Simulazioni da distribuzione Poisson

La funzione `simN` prende in input il campione x e il numero di simulazioni desiderate N e torna in output una lista composta da 3 matrici, corrispondenti ai metodi bootstrap, contenenti gli estremi degli intervalli di confidenza calcolati per le N simulazioni. La funzione contiene un ciclo `for` dentro cui è presente la funzione `sim.po` per la simulazione di un campione y da una distribuzione Poisson con x il vettore dei valori del campione `cloth` e coefficienti di regressione pari a 1.1 e 0.7. Su ciascun campione vengono applicate le funzioni `boot`, `boot.ci` e `ic.BC` per ottenere i risultati. Nell'utilizzo della funzione `boot.ci` si procede definendo la varianza come i valori ottenuti da `var.np`.

```
simN <- function(x,N){
```

```
out.np <- out.rp <- out.rd <- matrix(ncol=12, nrow=N)
colnames(out.np) <- colnames(out.rp) <- colnames(out.rd) <- c('norm.
 inf', 'norm.sup', 'perc.inf', 'perc.sup', 'base.inf', 'base.sup', '
 stud.inf', 'stud.sup', 'bca.inf', 'bca.sup', 'bc.inf', 'bc.sup')

for (i in 1:N){
  sim.po <- function(x, beta){
    x2 <- cbind(rep(1,length(x)),x)
    y <- rpois(n=length(x), lambda = exp(x2%*%beta))
    y
  }
  y <- sim.po(x,c(1.1,.7))
  mu.y <- mean(y)

  data.po <- glm(y~x, poisson, data=data.frame(y,x))
  data.diag <- glm.diag(data.po)
  data.rp <- data.frame(data.frame(y,x), fit=fitted(data.po),
                        pea=(data.diag$rp-mean(data.diag$rp)))

  if(min(y)==0) y[y==0] <- 0.001 ## se y=0 si ha rd.sim Nan
  y.sim <- seq(from=min(y), to=max(y), length.out=9999)
  rd.sim <- sign(y.sim-mu.y)*sqrt(2*(y.sim*log(y.sim/mu.y)
                                     -y.sim+mu.y))
  sim <- smooth.spline(rd.sim,y.sim)
  z <- predict(sim, data.diag$rd)$y
  data.rd <- data.frame(data.frame(y,x), z)

  ic.BC <- function(bootT){
    w <- qnorm(mean(bootT$t[,1] <= 1)) ## vero valore di phi = 1
    z <- qnorm(c(.025,.975))
    q_bc <- pnorm(2*w+z)
    quantile(bootT$t[,1], q_bc, names=F)
  }

  var.boot.np <- boot(data.frame(y,x), statistic = var.np, R=999)

  phi.boot.rp <- boot(data.rp, boot.fun.rp, R=999)
  ic.rp <- boot.ci(boot.out = phi.boot.rp, var.t = var.boot.np$t,
                   var.t0 = var.boot.np$t0)
  out.rp[i,] <- c(ic.rp$normal[,2:3], ic.rp$perc[,4:5],
                  ic.rp$basic[,4:5], ic.rp$stud[,4:5],
                  ic.rp$bca[,4:5], ic.BC(phi.boot.rp))

  phi.boot.rd <- boot(data.rd, boot.fun.rd, R=999)
```

```
    ic.rd <- boot.ci(boot.out = phi.boot.rd, var.t = var.boot.np$t,
                     var.t0 = var.boot.np$t0)
    out.rd[i,] <- c(ic.rd$normal[,2:3], ic.rd$perc[,4:5],
                    ic.rd$basic[,4:5], ic.rd$stud[,4:5],
                    ic.rd$bca[,4:5], ic.BC(phi.boot.rd))

    phi.boot.np <- boot(data.frame(y,x), statistic = phi.np, R=999)
    ic.np <- boot.ci(boot.out = phi.boot.np, var.t = var.boot.np$t,
                     var.t0 = var.boot.np$t0)
    out.np[i,] <- c(ic.np$normal[,2:3], ic.np$perc[,4:5],
                    ic.np$basic[,4:5], ic.np$stud[,4:5],
                    ic.np$bca[,4:5], ic.BC(phi.boot.np))
  }
  return(list('out.rp'=out.rp, 'out.rd'=out.rd, 'out.np'=out.np))
}
cov <- simN(cloth$x, N=1000)
```

Coperture empiriche

Per il calcolo delle coperture empiriche degli intervalli di confidenza si definisce la funzione `outside` che, dato il vero valore del parametro e la matrice degli estremi, torna la percentuale di intervalli di confidenza in cui non rientra il vero valore. La copertura empirica dell'intervallo di confidenza sarà il complemento ad 1 dell'output di tale funzione.

```
outside <- function(p, out){
  c((sum(p<out[,1])+sum(p>out[,2]))/length(out[,2]),(sum(p<out[,3])+
    sum(p>out[,4]))/length(out[,4]),(sum(p<out[,5])+sum(p>out[,6]))/
    length(out[,6]),(sum(p<out[,7])+sum(p>out[,8]))/length(out[,8]),
    (sum(p<out[,9])+sum(p>out[,10]))/length(out[,10]),(sum(p<out[,11])
    +sum(p>out[,12]))/length(out[,12]))
}

cov.rp <- 1-outside(1, cov$out.rp)
cov.rd <- 1-outside(1, cov$out.rd)
cov.np <- 1-outside(1, cov$out.np)
```

Simulazioni: $\phi > 1$

Funzioni bootstrap

Le funzioni per la definizione delle procedure bootstrap parametrica e non parametrica necessitano del caricamento di nuove librerie. La libreria `gamlss` permette la generazione e l'adattamento di dati dalla distribuzione binomiale negativa lineare, rispettivamente con le funzioni `rNBII` e `gamlss`. La funzione `rNBII` richiede i parametri `sigma`, corrispondente a τ, e la media `mu` delle osservazioni y. Nella funzione `gamlss` il modello di regressione NB1 è ottenuto con l'opzione `NBII` e restituisce la stima di τ e del suo standard error in scala logaritmica. La libraria `msm` contiene il comando `deltamethod` che permette la trasformazione dalla scala logaritmica a quella originale dello standard error utilizzando il metodo delta.

```
library(boot)
library(gamlss)
library(msm)

# parametrico
phi.nb1 <- function(data){
  nb <- summary(gamlss(y~x, family = NBII, data = data))[3,1:2]
  nb[1] <- exp(nb[1])
  nb[2] <- (deltamethod(~exp(x1), mean=nb[1], cov=(nb[2])^2))^2
  nb
}
sim.nb1 <- function(data, mle){
  d <- data
  d$y <- rNBII(n=nrow(data), sigma=mle[1], mu=mle[2])
  d
}

# non parametrico
phi.nb1.np <- function(data,i){
  d <- data[i,]
  nb <- summary(gamlss(y~x, family = NBII, data = d))[3,1:2]
  nb[1] <- exp(nb[1])
  nb[2] <- (deltamethod(~exp(x1), mean=nb[1], cov=(nb[2])^2))^2
  nb
}
```

Simulazioni da distribuzione NB1

La funzione `simN.nb1` ha struttura simile a `simN`, utilizzata in precedenza, prende in input un vettore di valori x e il numero N di simulazioni, e restituisce in output una lista contenente 2 matrici di dimensione Nx12, ciascuna composta dagli estremi degli intervalli di confidenza per ogni simulazione avvenuta. La simulazione dei valori dalla distribuzione NB1 avviene tramite la funzione `sim.nb` con veri valori dei parametri contenuti nel vettore $\beta = (2.5, 1.1, 0.7)$, dove il primo corrisponde a τ. Inoltre, all'interno del ciclo `for`, oltre alla funzione `ic.BC` ridefinita per $\tau = 2.5$, è presente la funzione `ic.BCa` per il calcolo di tale intervallo basato sul bootstrap parametrico con la forma approssimata con i valori jackknife.

```
simN.nb1 <- function(x,N){
  out.para <- out.np <- matrix(ncol=12, nrow=N)
  colnames(out.para) <- colnames(out.np) <- c('norm.inf', 'norm.sup',
   'perc.inf', 'perc.sup', 'base.inf', 'base.sup', 'stud.inf', 'stud.
   sup', 'bca.inf', 'bca.sup', 'bc.inf', 'bc.sup')

  for (i in 1:N){
    sim.nb <- function(x,beta){
      x2 <- cbind(rep(1,length(x)),x)
      y <- rNBII(n=length(x), sigma=beta[1], mu=exp(x2%*%beta[-1]))
      y
    }
    y <- sim.nb(x,c(2.5,1.1,.7))

    tau <- exp(summary(gamlss(y~x, family = NBII, data =
                                data.frame(y,x)))[3,1])
    mu.y <- mean(y)

    ic.BC <- function(bootT){
      w <- qnorm(mean(bootT$t[,1] <= 2.5))
      z <- qnorm(c(.025,.975))
      q_bc <- pnorm(2*w+z)
      quantile(bootT$t[,1], q_bc, names=F)
    }

    ic.BCa <- function(bootT){
      w <- qnorm(mean(bootT$t[,1] <= 2.5))
      z <- qnorm(c(.025,.975))
      n <- length(bootT$data$y)
      I <- rep(NA, n)
```

```
      for(i in 1:n){
        new <- bootT$data[-i,]
        jack <- exp(summary(gamlss(y~x, family = NBII, data = new))
                    [3,1])
        I[i] <- n*2.5-(n-1)*jack
      }
      a_hat <- (sum(I^3)/sum(I^2)^1.5)/6
      q_bca <- pnorm(w + (w+z)/(1-a_hat*(w+z)))
      quantile(bootT$t[,1], q_bca, names=F)
    }

    phi.para <- boot(data.frame(y,x), statistic = phi.nb1, R=999,
                sim='parametric', ran.gen = sim.nb1, mle=c(tau, mu.y))
    ic.para <- boot.ci(phi.para, type=c('norm','perc','basic','stud'))
    out.para[i,] <- c(ic.para$normal[,2:3], ic.para$perc[,4:5],
                      ic.para$basic[,4:5], ic.para$stud[,4:5],
                      ic.BCa(phi.para), ic.BC(phi.para))

    phi.boot.np <- boot(data.frame(y,x), statistic = phi.nb1.np,
                          R=999)
    ic.np <-  boot.ci(phi.boot.np)
    out.np[i,] <- c(ic.np$normal[,2:3], ic.np$perc[,4:5],
                    ic.np$basic[,4:5], ic.np$stud[,4:5],
                    ic.np$bca[,4:5], ic.BC(phi.boot.np))
  }
  return(list('out.para'=out.para, 'out.np'=out.np))
}
cov <- simN.nb1(cloth$x, N=1000)
```

Coperture empiriche

La valutazione riguardante le coperture empiriche avviene con la funzione `outside` definita nel paragrafo precedente.

```
cov.para <- 1-outside(2.5, cov$out.para)
cov.np <- 1-outside(2.5, cov$out.np)
```

Bibliografia

A. Canty & B. D. Ripley (2024). *boot: Bootstrap R (S-Plus) Functions*. R package version 1.3-30.

Agresti, A. (2015). *Foundations of Linear and Generalized Linear Models*. Wiley.

Bissell, A. F. (1972). A negative binomial model with varying element sizes. *Biometrika* **59**, 435–441.

Cochran, W. (1977). *Sampling Techniques, 3rd ed.* Wiley.

Davison, A. C. & Hinkley, D. V. (1997). *Bootstrap Methods and their Application*. Cambridge University Press.

Dunn, K. P. & Smyth, G. K. (1996). Randomized quantile residuals. *Journal of Computational and Graphical Statistics* **5**, 264–282.

Hilbe, J. M. (2011). *Negative Binomial Regression, 2nd ed.* Cambridge University Press.

Lawless, J. F. (1987). Negative binomial and mixed poisson regression. *Canadian Journal of Statistics* **15**, 209–225.

McCullagh, P. & Nelder, J. A. (1989). *Generalized Linear Models, 2nd ed.* Chapman and Hall.

Pace, L. & Salvan, A. (1996). *Teoria della statistica: Metodi, Modelli, Approssimazioni Asintotiche*. Cedam.

Rao, C. R. & Chakravarti, I. M. (1956). Some small sample tests of significance for a poisson distribution. *Biometrics* **12**, 264–282.

Salvan, A., Sartori, N. & Pace, L. (2020). *Modelli lineari generalizzati*. Springer.

Sartori, N. & Guolo, A. (2022). *Materiale del corso di Statistica Computazionale progradito, A.A. 2022-2023*.