Politecnico di Torino

Politecnico di Torino

Ingegneria Matematica

A.a. 2024/2025

Sessione di laurea Luglio 2025

Metodi statistici per il confronto dei tassi di eventi avversi di trattamenti clinici alternativi

Relatori:

Mauro Gasparini

Candidati:

Vito La Piana

Indice

Sommario

Questa tesi si focalizza sull'implementazione del metodo BDRIBS (Bayesian Detection of potential Risk using Inference on Blinded Safety data) esplorando il suo comportamento sotto diverse configurazioni di parametri. Successivamente, viene proposto un nuovo metodo basato su un approccio frequentista, in alternativa al tradizionale approccio bayesiano. Verrà analizzata la struttura di entrambi i modelli, i loro risultati saranno simulati, commentati e infine comparati tra di loro. Questi dimostrano che il modello frequentista non solo mantiene un elevato livello di affidabilità, ma offre anche vantaggi in termini di semplicità computazionale e interpretabilità.

Capitolo 1

Introduzione

La situazione oggetto di studio in questa tesi riguarda l'analisi di un nuovo farmaco o, più in generale, di un nuovo trattamento, con l'obiettivo di valutare se si abbia una così alta evidenza empirica che il rischio relativo di eventi avversi sia maggiore di uno, tale da giustificare la cessazione della sperimentazione in cieco. Si desidera quindi stabilire se il rischio di sviluppare un evento avverso sia superiore per i pazienti sottoposti al trattamento sperimentale rispetto a quelli sottoposti al trattamento di controllo, anche quando si abbiano a disposizione informazioni parziali. Questa analisi è fondamentale in fase di sperimentazione clinica, poichè consente di bilanciare l'efficacia di un trattamento con la sua sicurezza.
Durante la sperimentazione a ciascun paziente coinvolto nello studio viene assegnato casualmente un trattamento, con una probabilità determinata da un fattore k, che rappresenta il rapporto tra la probabilità di essere assegnati al gruppo di trattamento sperimentale e quella di essere assegnati al gruppo di controllo. Tale assegnazione randomizzata è fondamentale per garantire l'imparzialità e la validità statistica dei risultati, minimizzando potenziali bias legati a fattori esterni.
Durante il periodo di osservazione, al paziente può succedere una delle seguenti due possibilità: può verificarsi un evento avverso; oppure il paziente può essere censurato, ovvero l'osservazione si interrompe prima che si manifesti un evento. Questo fenomeno viene trattato, separatamente per gruppo di controllo e gruppo sperimentale, modellando un processo di osservazione che considera gli eventi avversi rilevati, e che tratta i censuramenti come eventi avversi non avvenuti. Una spiegazione più accurata verrà fornita nei capitoli successivi.
Tuttavia, nella realtà degli studi clinici, i dati sono frequentemente raccolti in cieco. Questo accade principalmente per l'esigenza di garantire l'integrità dello studio clinico e per minimizzare il rischio di influenze esterne sui risultati. In particolare, gli sponsor degli studi possono optare per una revisione in cieco dei dati di sicurezza per soddisfare specifiche esigenze normative e operative. [1] La Food and Drug Administration (FDA) raccomanda che un Safety Assessment

Committee (SAC) interno conduca revisioni aggregate non cieche dei dati, ma queste analisi non cieche di routine possono comportare problematiche significative, come il rischio di un inavvertito scoprimento dei dati o anche solo l'apparenza di una violazione dell'imparzialità da parte del personale coinvolto nello studio. Di conseguenza, le informazioni disponibili sono spesso limitate al numero totale di eventi avversi osservati, senza la possibilità di sapere quanti appartengano al gruppo di trattamento e quanti al gruppo di controllo. Questa limitazione rappresenta una sfida significativa, che richiede l'adozione di tecniche avanzate per estrapolare informazioni affidabili dai pochi dati disponibili.

Capitolo 2

Analisi del metodo BDRIBS

BDRIBS [2] è un metodo statistico sviluppato presso AbbVie per supportare il *Safety Management Team* (SMT) nel monitoraggio e nella valutazione dei dati di sicurezza in cieco durante uno studio clinico in corso. Il metodo modella il rischio relativo (r) di un farmaco sperimentale rispetto al controllo, considerando una singola categoria di eventi avversi alla volta.

2.1 Ipotesi del modello

Si assume che il tasso di incidenza per anno-paziente sia piccolo e costante nel tempo. Queste ipotesi consentono di modellare il numero di eventi in ciascun gruppo come un processo di Poisson omogeneo. Inoltre, si presuppone che sia disponibile una stima del tasso storico di eventi di base per il braccio di controllo. Le ipotesi fondamentali del modello sono le seguenti:

$$y_e \sim \text{Poisson}(\lambda_e E_e), \quad y_c \sim \text{Poisson}(\lambda_c E_c),$$
$$r = \frac{\lambda_e}{\lambda_c}$$

dove y_e e y_c rappresentano rispettivamente il numero di eventi avversi nei gruppi di trattamento e di controllo; λ_e e λ_c sono i tassi unitari dei processi di Poisson per i rispettivi gruppi; E_e e E_c indicano i tempi di osservazione espressi in anni-paziente. Inoltre, avendo definito k come il rapporto tra la probabilità di essere assegnati al gruppo sperimentale e la probabilità di essere assegnati al gruppo di controllo, si assume che:

$$E_e \approx kE_c$$

Approssimando la similitudine ad un'uguaglianza abbiamo che

$$E = E_c + E_e = E_c + kE_c \implies E_c = \frac{E}{1+k}$$

Da cui segue che

$$E_c\lambda_c + E_e\lambda_e = E_c\lambda_c + E_e r\lambda_c = \lambda_c(E_c + rE_e) = \lambda_c(E_c + rkE_c) =$$
$$= \lambda_c E_c(1 + rk) = \lambda_c E\frac{1 + rk}{1 + k}$$

Il modello sarà quindi rappresentato da

$$y \sim \text{Poisson}(\lambda_c E\frac{1 + rk}{1 + k})$$

dove y è il numero totale di eventi avversi osservati; sebbene apparentemente possa sembrare un singolo dato, y in realtà racchiude informazioni su numerosi pazienti, essendo il risultato del conteggio di tutti gli eventi avversi.
Nel modello, λ_c viene considerato noto in quanto stimato a partire da dati storici; in alternativa, gli si può assegnare una distribuzione a priori, che giustifica l'impostazione bayesiana del metodo, in quanto si cerca di sfruttare al massimo l'informazione disponibile a priori. E e k sono noti, mentre l'unica incognita è il rischio relativo r.

2.2 Scelta della distribuzione a priori

Poichè stiamo assumendo di non avere alcuna informazione riguardo a rischi addizionali per il gruppo di trattamento rispetto al gruppo di controllo, è appropriato assegnare al rischio relativo una distribuzione a priori che sia il più possibile non informativa. Si definisce p come la probabilità che, dato un evento avverso, esso sia associato al gruppo di trattamento. Nel linguaggio matematico:

$$y_e \mid y \sim \text{Binomiale}(y, p)$$

dove

$$p = \frac{\lambda_e}{\lambda_c + \lambda_e} = \frac{kr}{kr + 1}$$

da cui

$$r = \frac{p}{k(1 - p)}$$

con p definito nell'intervallo $[0, 1]$, al quale viene assegnata una distribuzione a priori non informativa. Nel caso specifico in cui $k = 1$, si assume:

$$p \sim \text{Uniforme}(0,1)$$

che equivale a

$$p \sim \text{Beta}(1,1)$$

da cui segue che

$$r \sim \text{Inversa Beta}(1,1)$$

Questo tipo di scelta non va bene nel caso in cui $k \neq 1$, poichè darebbe una informazione a priori di maggiore rischio del trattamento sperimentale nel caso in cui $k < 1$, e di minore rischio nel caso in cui $k > 1$. Un'analisi per stabilire quale distribuzione utilizzare in questi due casi verrà fatta nei capitoli successivi.

2.3 Output del modello

Il modello bayesiano produce come risultato una distribuzione a posteriori del parametro r. L'interesse principale si concentra sulla quantità:

$$P(r > r_o \mid y)$$

Nel presente lavoro si considera $r_o = 1$, ma in generale può assumere anche altri valori. Se tale probabilità supera una certa soglia predeterminata, viene lanciato un allarme e si chiede la scopertura dei dati, cioè la cessazione del cieco. In caso contrario, non vi è evidenza per affermare che il trattamento sperimentale sia più pericoloso rispetto al controllo.
Chiaramente diventa cruciale la scelta della soglia di allarme da adottare. In particolare essa deve adattarsi alle esigenze della situazione; se vogliamo evitare maggiormente il rischio di falso positivo, ovvero di lanciare l'allarme quando non necessario, allora bisogna adottare una soglia particolamente elevata; al contrario, se si è più interessati a non avere un falso negativo, ovvero non lanciare l'allarme quando $r > 1$, allora bisogna utilizzare una soglia di allarme un pò più bassa.

Capitolo 3

Processo di osservazione

Non disponendo di dati reali, utilizziamo dati simulati su R per testare il modello BDRIBS. Il primo passo della simulazione consiste nella costruzione di un processo di punti, che modella i casi avversi come arrivi di un processo di Poisson. In questo capitolo, omettiamo i pedici e e c che identificano i due gruppi di trattamento.

3.1 Costruzione del processo di osservazione

Avendo assunto che i casi avversi seguano un processo di Poisson con parametro λ, il tempo di arrivo di un singolo caso avverso per un paziente è distribuito come una variabile esponenziale di parametro λ. Tuttavia, per motivi pratici, ogni paziente viene osservato solo per un periodo limitato: il tempo di osservazione termina con un censuramento, che in questa tesi viene modellato come una variabile esponenziale di parametro λ_{cens}.
Consideriamo N pazienti. Per ciascuno i tempi di arrivo degli eventi avversi $x_1, x_2, \ldots, x_N$ sono campionati come variabili i.i.d. da una distribuzione esponenziale con parametro λ; i tempi di censuramento $c_1, c_2, \ldots, c_N$ sono campionati come variabili i.i.d. da una distribuzione esponenziale con parametro λ_{cens}

$$x_1, x_2, \ldots, x_N \text{ i.i.d.} \sim \text{Exp}(\lambda)$$

$$c_1, c_2, \ldots, c_N \text{ i.i.d.} \sim \text{Exp}(\lambda_{\text{cens}})$$

Questi due insieme di variabili aleatorie non sono osservabili, mentre quello che si riesce ad osservare sono i tempi $t_1, t_2, \ldots t_n$ definiti come

$$t_i = \min(x_i, c_i), \quad \forall i \in \{1, \ldots, N\}$$

Per ogni paziente, definiamo inoltre un indicatore binario δ_i che ci dice se i tempi osservati corrispondono ad un evento avverso oppure no

$$\delta_i = \begin{cases} 0, & \text{se } t_i = c_i, \\ 1, & \text{se } t_i = x_i \end{cases}$$

Supponiamo di avere a disposizione unicamente i valori t_i e δ_i per $i = 1, \ldots, N$, come avviene nei casi reali. L'operazione che viene fatta consiste nel costruire un processo di punti in cui i tempi intercorrenti siano la somma dei tempi censurati consecutivi $t_i : \delta_i = 0$ e il primo tempo non censurato $t_j : \delta_j = 1$. Questo calcolo viene ripetuto per tutta la sequenza di dati osservati, ricominciando una nuova sequenza ogni volta che si incontra un evento avverso.

3.2 Logica e formalità del processo

Dal punto di vista logico, questo procedimento permette di considerare solo gli arrivi osservabili, ossia quelli che avvengono prima del censuramento. I tempi censurati vengono sommati al tempo del successivo arrivo osservato, poiché, dal punto di vista di chi conduce lo studio, un evento avverso censurato equivale a un evento "non osservato".
Formalmente, si costruisce un vettore $\boldsymbol{\tau} = (\tau_1, \tau_2, \ldots, \tau_y)$, con

$$\tau_k = \sum_{i=n_k}^{m_k} t_i,$$

dove n_k è l'indice del primo tempo in una sequenza di censuramenti consecutivi e m_k è l'indice del primo tempo successivo tale che $\delta_{m_k} = 1$.
Direttamente da $\boldsymbol{\tau}$ ricaveremo il numero di eventi avversi, uguale alla dimensione del vettore, e il tempo totale di osservazione, uguale alla somma di tutte le componenti del vettore.
Tutto il procedimento viene applicato separatamente al gruppo di controllo e al gruppo sperimentale.
Si può dimostrare che i tempi intercorrenti sono indipendenti dal valore di λ_{cens}, e che sono distribuiti come un'esponenziale di parametro λ; questo ci garantisce che ha senso simulare il processo di osservazione in questo modo per la nostra analisi.

3.3 Analisi della variazione dei parametri simulati

È stato studiato come variano i parametri del vettore $\boldsymbol{\tau}$ al variare dei valori di N, λ e λ_{cens}. In particolare, ci interessano, come spiegato precedentemente, il numero di arrivi avversi osservati, indicato con y, il tempo di osservazione espresso in anni-paziente, indicato con E, e la media delle componenti del vettore $\boldsymbol{\tau}$, che

rappresenta una stima del valore atteso dei tempi intercorrenti.
Poiché i risultati ottenuti sono soggetti ad aleatorietà, ogni configurazione di parametri è stata sottoposta a 500 simulazioni, calcolandone successivamente la media delle quantità di interesse.
In Tabella 3.1 sono riportati i risultati ottenuti:

Tabella 3.1

N	λ	λ_{cens}	y	E	$\bar{\tau}_k$
100	0.01	0.08	11.97	1111.61	99.99
200	0.01	0.08	23.11	2222.09	100.04
500	0.01	0.08	56.41	5500.83	100.03
1000	0.01	0.08	112.01	11109.29	99.97
1000	0.02	0.08	200.90	10000.36	44.99
1000	0.05	0.08	385.13	7691.53	19.98
1000	0.1	0.08	555.85	5556.90	100.01
1000	0.01	0.16	59.77	5882.84	100.10
1000	0.01	0.5	20.59	1961.81	99.98
1000	0.01	1	10.97	990.38	100.12

I risultati sono coerenti con quanto affermato nel paragrafo precedente, ovvero che i tempi intercorrenti sono distribuiti secondo una distribuzione esponenziale con parametro λ. Infatti, sappiamo che se

$$T \sim \text{Exp}(\lambda), \quad \text{allora} \quad \mathbb{E}[T] = \frac{1}{\lambda},$$

che è approssimativamente ciò che osserviamo nelle simulazioni per la media delle componenti del vettore.
Si osserva inoltre che all'aumentare del numero di pazienti N, sia il numero di arrivi y sia il tempo di osservazione E crescono in modo direttamente proporzionale.
Sappiamo inoltre, per le proprietà della distribuzione esponenziale, che

$$\mathbb{P}(x < c) = \frac{\lambda}{\lambda + \lambda_{\text{cens}}},$$

quindi all'aumentare di λ aumentano le probabilità che $\delta_i = 1$, mentre all'aumentare di λ_{cens} esse diminuiscono. Questo spiega perché y cresce con λ e diminuisce con λ_{cens}.
Infine, sempre per le proprietà della distribuzione esponenziale, sappiamo che

$$\mathbb{E}[\min(x, c)] = \frac{1}{\lambda + \lambda_{\text{cens}}},$$

quindi sia un aumento di λ che un aumento di λ_{cens} comportano una diminuzione della media dei t_i, e di conseguenza una riduzione del tempo totale di osservazione. Alla fine di questa analisi possiamo concludere che tutto quello che aspettavamo è stato confermato dai risultati empirici.

3.4 Codice R

Di seguito il codice implementato per simulare il vettore $\boldsymbol{\tau}$:

```
1
2  # Numero di pazienti
3  N <- 10000
4
5  # Definiamo i rate di Poisson
6  lambda <- 0.1
7  lambdacens <- 0.08
8
9  # Simuliamo il vettore dei tempi di casi avversi
10 x <- rexp(N, rate = lambda)
11
12 # Simuliamo il vettore dei tempi di censuramento
13 c <- rexp(N, rate = lambdacens)
14
15 # Creiamo il vettore dei tempi osservabili
16 t <- pmin(x, c)
17
18 # Creiamo il vettore degli indicatori binari
19 delta <- ifelse(t == c, 0, 1)
20
21 # Creiamo un vettore vuoto per i "tacconi"
22 tau <- c()
23
24 # Iniziamo a scorrere il vettore delta
25 i <- 1
26 while (i <= N) {
27   # Variabile che serve  per il calcolo del taccone
28   sum_censored <- 0
29
30   # Somma dei valori censurati consecutivi
31   while (i <= N && delta[i] == 0) {
32     sum_censored <- sum_censored + t[i]
33     i <- i + 1
34   }
35
36   # Aggiungiamo il primo valore non censurato
37     if (i <= N && delta[i] == 1) {
38       sum_censored <- sum_censored + t[i]
39     }
```

```
40      tau <- c(tau, sum_censored)
41
42    # Passa al prossimo elemento
43    i <- i + 1
44  }
```

Questo codice sarà integrato per due volte (una per il gruppo di controllo e una per il gruppo sperimentale) in uno script più ampio quando andremo a testare il comportamento del metodo BDRIBS, variando i valori di λ. E' stato assegnato un valore arbitrario a λ_{cens}, ma ciò risulta irrilevante per quanto riguarda l'esponenzialità del vettore $\boldsymbol{\tau}$, che è garantita qualunque esso sia.

Capitolo 4

Simulazione del metodo BDRIBS

Dopo aver allocato casualmente i pazienti tra il gruppo di controllo e il gruppo sperimentale (in questa fase ci concentriamo sul caso $k = 1$) e dopo aver simulato il processo di osservazione per entrambi i gruppi, otteniamo le quantità y_c,y_e,E_c,E_e. Tuttavia, per usare le simulazioni, fingiamo di non conoscere separatamente il numero di eventi avversi per il controllo e per il trattamento, ma solo la loro somma complessiva. Allo stesso modo, fingiamo di non conoscere il tasso di incidenza del gruppo sperimentale e di avere a disposizione esclusivamente il tasso del gruppo di controllo.
Per testare il modello, considereremo diversi valori del tasso di incidenza e del numero totale di pazienti (N). Per ogni configurazione, eseguiremo numerose simulazioni, con l'obiettivo di valutare le prestazioni del metodo. In particolare, il modello svolge una funzione di classificazione: segnala un allarme quando rileva un rischio relativo superiore alla soglia prefissata ("positivo") oppure decide di non lanciare l'allarme ("negativo").

$$P(r > r_o \mid y) > soglia \implies \text{allarme}$$

$$P(r > r_o \mid y) < soglia \implies \text{non allarme}$$

Utilizzando i dati simulati come input del modello, ci interessa contare quante volte il modello classifica correttamente e quante volte no, per valutare le probabilità di successo e di insuccesso.

4.1 Prestazioni del metodo con $k = 1$

E' stato scelto di ripetere le simulazioni 500 volte per ogni caso; i risultati ottenuti sono riportati in Tabella 4.1

Si può vedere che per $N = 50$ e $\lambda_c = 0.01$ l'utilizzo della soglia pari a 0.8 sia abbastanza equilibrato, e in entrambi i casi porta a prestazioni del modello di circa l'80%.
Successivamente è stato aumentato il numero di pazienti, e di conseguenza è stato aumentato anche il numero di eventi avversi ed il tempo di osservazione. Quello che succede è piuttosto interessante: come si vede dall'immagine, nel caso in cui $r = 1$ le prestazioni del modello restano praticamente identiche; al contrario, nel caso in cui il rate del nuovo trattamento è il doppio rispetto al rate del controllo, il modello si comporta in maniera perfetta, rilevando in tutte le simulazioni effettuate la "positività". E' ragionevole pensare quindi di aumentare la soglia, per forzare la decisione verso "negativo" e aumentare le prestazioni del primo caso. Anche utilizzando una soglia molto elevata, ovvero del 99% le prestazioni restano perfette nel secondo caso, e come ci aspettavamo, aumentano notevolmente nel primo.
Nell'ultima simulazione viene diminuito nuovamente il numero di pazienti, aumentando però il rate unitario di casi avversi; cosi facendo viene aumentato il numero di eventi avversi, quindi ci si aspetta un comportamento simile a quello precedente; tuttavia i risultati dicono che questo non succede. Questo è dovuto al fatto che, come visto nei capitoli precedenti, un aumento del rate unitario porta anche ad una riduzione del tempo di osservazione; quindi il rate di arrivi effettivo, essendo direttamente proporzionale sia al rate unitario sia al tempo di osservazione, non subirà un aumento.

Tabella 4.1: Risultati sperimentali metodo bayesiano

k	N	soglia	λ_c	λ_e	r	TN	FP	TP	FN
1	50	0.8	0.01	0.01	1	407	93	/	/
1	50	0.8	0.01	0.02	2	/	/	403	97
1	500	0.8	0.01	0.01	1	404	96	/	/
1	500	0.8	0.01	0.02	2	/	/	500	0
1	500	0.99	0.01	0.01	1	495	5	/	/
1	500	0.99	0.01	0.02	2	/	/	500	0
1	50	0.8	0.1	0.1	1	407	93	/	/
1	50	0.8	0.1	0.2	2	/	/	411	89

4.2 Codice R

Di seguito il codice implementato per testare il modello e ottenere i risultati della prima riga della Tabella 4.1 Per quanto riguarda gli altri casi bisognerà solo cambiare i valori numerici.

```
1 library(R2jags)
```

```
2  library(ggplot2)
3
4  # Rate di casi avversi di controllo ed  esperimento
5  lambda_c <- 0.01
6  lambda_e <- 0.01
7
8  # Soglia che verrà utilizzata per stabilire se lanciare un allarme
9  treshold <- 0.8
10
11 # Numero di simulazioni
12 num_simulations <- 500
13
14 # Vettore per salvare le probabilità che r > 1
15 prob_r_gt_1_list <- numeric(num_simulations)
16 h <- 1
17 # Ciclo per ripetere la simulazione
18 for (h in 1:num_simulations) {
19
20   # Definiamo il numero di pazienti totali
21   N <- 50
22
23   # Allochiamo in maniera random ed equiprobabile i pazienti tra
       gruppo di controllo e gruppo sperimentale
24   groups <- sample(c("c", "e"), size = N, replace = TRUE, prob = c
       (0.5, 0.5))
25
26   # Contiamo il numero di pazienti per ogni gruppo
27   count_c <- sum(groups == "c")
28   count_e <- sum(groups == "e")
29
30   # Creiamo il processo di osservazione del gruppo controllo
31   x_c <- rexp(count_c, rate = lambda_c)
32   c_c <- rexp(count_c, rate = 0.008)
33   t_c <- pmin(x_c, c_c)
34   delta_c <- ifelse(t_c == c_c, 0, 1)
35   tau_c <- c()
36   i <- 1
37   while (i <= count_c) {
38     sum_censored_c <- 0
39     censored_count_c <- 0
40     while (i <= count_c && delta_c[i] == 0) {
41       sum_censored_c <- sum_censored_c + t_c[i]
42       i <- i + 1
43     }
44     if (i <= count_c && delta_c[i] == 1) {
45       sum_censored_c <- sum_censored_c + t_c[i]
46     }
47     tau_c <- c(tau_c, sum_censored_c)
48     i <- i + 1
```

```
  }


  # Creiamo il processo di osservazione del gruppo sperimentale
  x_e <- rexp(count_e, rate = lambda_e)
  c_e <- rexp(count_e, rate = 0.008)
  t_e <- pmin(x_e, c_e)
  delta_e <- ifelse(t_e == c_e, 0, 1)
  tau_e <- c()
  i <- 1
  while (i <= count_e) {
    sum_censored_e <- 0
    censored_count_e <- 0
    while (i <= count_e && delta_e[i] == 0) {
      sum_censored_e <- sum_censored_e + t_e[i]
      i <- i + 1
    }
    if (i <= count_e && delta_e[i] == 1) {
      sum_censored_e <- sum_censored_e + t_e[i]
    }
    tau_e <- c(tau_e, sum_censored_e)
    i <- i + 1
  }


  # Ricaviamo il numero di arrivi di controllo ed esperimento
  y_c <- length(tau_c)
  y_e <- length(tau_e)

  # Sommiamo e ricaviamo il numero di arrivi totali
  y <- y_c + y_e

  # Ricaviamo i tempi di osservazione di esperimento e controllo
  E_c <- sum(tau_c)
  E_e <- sum(tau_e)


  # Modello JAGS
  model_code <- "
  model {
    y ~ dpois((E_c + E_e) * lambda_c * (1 + r) / 2)
    r <- p / (1 - p)
    p ~ dunif(0, 1)
  }
  "

  # Dati per JAGS
  model_data <- list(y = y, E_c = E_c, E_e = E_e, lambda_c = lambda
    _c)
```

```
97
98    # Parametri da salvare
99    model_params <- c("p", "r")
100
101   # Esecuzione del modello
102   model_run <- jags(
103     data = model_data,
104     parameters.to.save = model_params,
105     model.file = textConnection(model_code),
106     n.chains = 2,
107     n.burnin = 500,
108     n.iter = 2000,
109     quiet = TRUE
110   )
111
112   # Estraiamo i campioni posteriori di r
113   r_samples <- model_run$BUGSoutput$sims.list$r
114
115   # Calcoliamo la probabilità che r sia maggiore di 1
116   prob_r_gt_1_list[h] <- mean(r_samples > 1)
117 }
118
119 # Stampiamo i risultati delle simulazioni
120 print(prob_r_gt_1_list)
121
122 # Contiamo quante volte le probabilità sono maggiori della soglia
123 count_greater_than_treshold <- sum(prob_r_gt_1_list > treshold)
124 count_less_or_equal_treshold <- sum(prob_r_gt_1_list <= treshold)
125
126 # Stampiamo i risultati
127 print(paste("Probabilità r > 1 maggiore della soglia: ", count_
      greater_than_treshold))
128 print(paste("Probabilità r > 1 minore della soglia: ", count_less_
      or_equal_treshold))
```

4.3 Simulazione del metodo nel caso in cui $k \neq 1$

Come anticipato nei capitoli precedenti, quando $k \neq 1$ occorre adottare una diversa scelta per la distribuzione a priori. Per garantire che non vengano date informazioni a priori sul rischio del trattamento sperimentale rispetto al trattamento standard, è necessario che

$$P(r > 1) = P(r < 1) = 0.5$$

Dato che p rappresenta una probabilità, è definito nell'intervallo [0,1]; la scelta naturale della distribuzione da assegnargli è quindi una $Beta(\alpha, \beta)$.

Nel caso in cui $k = 2$ (quando il paziente viene assegnato al trattamento sperimentale con probabilità $\frac{2}{3}$ e al trattamento tradizionale con probabilità $\frac{1}{3}$) vogliamo che

$$\mathbb{E}[p] = \frac{2}{3} \implies \frac{\alpha}{\alpha+\beta} = \frac{2}{3} \implies \alpha = 2\beta$$

In questo modo è stato imposto un vincolo, per imporre il secondo vincolo l'idea è stata quella di provare a massimizzare la varianza

$$\text{Var}(p) = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)} = \frac{2\beta^2}{9\beta^2(3\beta+1)} = \frac{2\beta^2}{27\beta^3+9\beta^2} = \frac{2}{27\beta+9}$$

$$\implies \frac{\partial \text{Var}(p)}{\partial\beta} = \frac{-54\beta}{(27\beta+9)^2}$$

Dalla derivata si vede che la funzione varianza si massimizza per $\beta = 0$, valore non accettabile. Più ci si avvicina allo 0 più si aumenta la varianza; tuttavia valori troppo piccoli di α e di β porterebbero ad una distribuzione "estrema", che produce solo valori prossimi a 0 e ad 1.E' stato scelto di utilizzare una $Beta$(0.2,0.1) come distribuzione a priori di p.
Se invece $k = \frac{1}{2}$ (quando il paziente viene assegnato al trattamento tradizionale con probabilità $\frac{2}{3}$ e al trattamento sperimentale con probabilità $\frac{1}{3}$) vogliamo che

$$\mathbb{E}[p] = \frac{1}{3} \implies \frac{\alpha}{\alpha+\beta} = \frac{1}{3} \implies \beta = 2\alpha$$

Seguendo un ragionamento analogo al caso precedente, è stata scelta una $Beta$(0.1,0.2).
In figura 4.1 le funzioni densità di probabilità scelte come priori

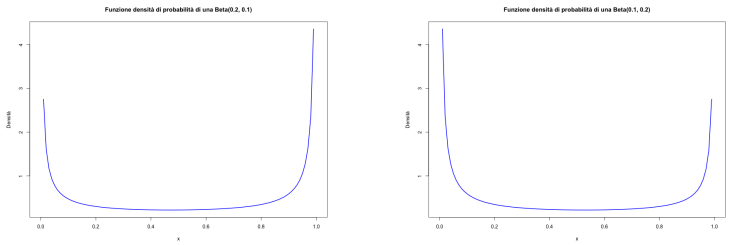

Figura 4.1

In Tabella 4.2 sono mostrati i risultati. Le prestazioni del modello nel primo caso sono decisamente migliori, come era lecito aspettarsi. Questo risultato è abbastanza ovvio, poiché nel nostro modello le informazioni sul trattamento sperimentale, che rappresenta il principale oggetto di studio, rivestono un ruolo più rilveante

rispetto a quelle sul trattamento tradizionale. Di conseguenza, all'aumentare di k, le prestazioni del modello migliorano.
Per ottenere un equilibrio tra falsi positivi e falsi negativi, sono state adottate soglie differenti. In particolare, per $k = 2$ la soglia è stata aumentata, mentre per $k = 0.5$ è stata ridotta. Questo risultato suggerisce che all'aumentare del numero di pazienti nel gruppo sperimentale il modello migliora soprattutto nella capacità di rilevare i casi di positività.

Tabella 4.2: Risultati sperimentali metodo bayesiano

k	N	soglia	λ_c	λ_e	r	TN	FP	TP	FN
2	50	0.85	0.01	0.01	1	441	59	/	/
2	50	0.85	0.01	0.02	2	/	/	452	48
2	500	0.99	0.01	0.01	1	498	2	/	/
2	500	0.99	0.01	0.02	2	/	/	500	0
0.5	50	0.7	0.01	0.01	1	365	135	/	/
0.5	50	0.7	0.01	0.02	2	/	/	338	164
0.5	500	0.97	0.01	0.01	1	480	20	/	/
0.5	500	0.97	0.01	0.02	2	/	/	478	22

Capitolo 5

Metodo frequentista

La parte innovativa di questo articolo consiste nell'introduzione di un metodo alternativo al BDRIBS, che abbia gli stessi obiettivi nello stesso contesto, adottando però un approccio frequentista.

5.1 Struttura del modello

Anche in questo caso, dopo aver simulato il processo di osservazione per il gruppo di controllo e quello sperimentale, vengono forniti in input al modello il numero totale di eventi avversi, il rate di incidenza del gruppo di controllo e i tempi di osservazione per entrambi i gruppi.
Il modello, come il precedente, effettua una classificazione: restituisce un esito "positivo" o "negativo" a seconda che ci sia o meno evidenza statistica che suggerisca che il trattamento nuovo è più pericoloso rispetto a quello standard.
L'ipotesi nulla afferma che il rate di incidenza del trattamento sperimentale coincida con quello del trattamento di controllo, e che quindi il numero di casi avversi totali rilevati sia distribuito come una Poisson di parametro uguale al prodotto tra λ_c e la somma dei tempi di osservazione dei due gruppi. L'ipotesi alternativa afferma invece che il trattamento sperimentale sia più pericoloso rispetto a quello tradizionale:

$$H_0 : \lambda_e = \lambda_c \implies y \sim \text{Poisson}(\lambda_c(E_c + E_e))$$

$$H_1 : \lambda_e > \lambda_c$$

Il metodo frequentista considera il p-value dell'ipotesi nulla contro l'ipotesi alternativa, che sarà uguale alla probabilità complementare della funzione di ripartizione della distribuzione dell'ipotesi nulla, calcolata nella y che viene data in input:

$$p\text{-value} = 1 - F_Y(y) = 1 - P(Y \leq y) = 1 - \sum_{k=0}^{y} \frac{e^{-\lambda_c(E_c+E_e)}(\lambda_c(E_c + E_e))^k}{k!}$$

dove

$$Y \sim \text{Poisson}(\lambda_c(E_c + E_e))$$

Se il p-value è più grande di una soglia prestabilita si accetta l'ipotesi nulla, altrimenti si accetta l'ipotesi alternativa e si lancia l'allarme per scoprire i dati:

$$p\text{-value} > soglia \implies \text{non allarme}$$

$$p\text{-value} < soglia \implies \text{allarme}$$

Valori bassi della soglia favoriranno il rischio di "falso negativo", valori più alti invece favoriranno il rischio di "falso positivo".

5.2 Prestazioni del modello

Al fine di fare un confronto tra questo modello ed il BDRIBS, i casi analizzati sono gli stessi analizzati precedentemente, e anche questa volta vengono riportati i risultati ottenuti con una soglia "neutra", ovvero una soglia che non privilegia nè la rilevazione della positività nè la rilevazione della negatività. Guardando la Tabella 5.1, che riporta i conteggi delle classificazioni nel caso di equiprobabilità tra controllo ed esperimento, si vede subito come anche questo metodo sia valido e i risultati ottenuti siano soddisfascenti.

Tabella 5.1: Risultati sperimentali metodo frequentista

k	N	soglia	λ_c	λ_e	r	TN	FP	TP	FN
1	50	0.12	0.01	0.01	1	405	95	/	/
1	50	0.12	0.01	0.02	2	/	/	403	97
1	500	0.12	0.01	0.01	1	432	68	/	/
1	500	0.12	0.01	0.02	2	/	/	500	0
1	500	0.01	0.01	0.01	1	497	3	/	/
1	500	0.01	0.01	0.02	2	/	/	500	0
1	50	0.12	0.1	0.1	1	417	83	/	/
1	50	0.12	0.1	0.2	2	/	/	414	86

In Tabella 5.2 sono invece riportate le prestazioni ottenute per $k \neq 1$, anche queste assolutamente in linea con quelle dell'altro metodo.

Tabella 5.2: Risultati sperimentali metodo frequentista

k	N	soglia	λ_c	λ_e	r	TN	FP	TP	FN
2	50	0.06	0.01	0.01	1	452	48	/	/
2	50	0.06	0.01	0.02	2	/	/	454	46
2	500	0.005	0.01	0.01	1	495	5	/	/
2	500	0.005	0.01	0.02	2	/	/	500	0
0.5	50	0.2	0.01	0.01	1	359	141	/	/
0.5	50	0.2	0.01	0.02	2	/	/	353	147
0.5	500	0.03	0.01	0.01	1	484	16	/	/
0.5	500	0.03	0.01	0.02	2	/	/	484	16

5.3 Codice R

Di seguito il codice implementato per testare il modello e ottenere i risultati della prima riga della Tabella 5.1 Per quanto riguarda gli altri casi bisognerà solo cambiare i valori numerici.

```
1  library(ggplot2)
2
3  # Rate di casi avversi di controllo ed esperimento
4
5  lambda_c <- 0.01
6  lambda_e <- 0.01
7
8  # Soglia che verrà utilizzata per stabilire se lanciare un allarme
9  treshold <- 0.12
10
11 # Numero di simulazioni
12 num_simulations <- 500
13
14 # Vettore per salvare le probabilità che r > 1
15 pvalue_list <- numeric(num_simulations)
16 h <- 1
17 # Ciclo per ripetere la simulazione
18 for (h in 1:num_simulations) {
19
20   # Definiamo il numero di pazienti totali
21   N <- 50
22
23   # Allochiamo in maniera random ed equiprobabile i pazienti tra
       gruppo di controllo e gruppo sperimentale
24   groups <- sample(c("c", "e"), size = N, replace = TRUE, prob = c
       (0.5, 0.5))
25
```

```
26  # Contiamo il numero di pazienti per ogni gruppo
27  count_c <- sum(groups == "c")
28  count_e <- sum(groups == "e")
29
30  # Creiamo il processo di osservazione del gruppo di controllo
31  x_c <- rexp(count_c, rate = lambda_c)
32  c_c <- rexp(count_c, rate = 0.008)
33  t_c <- pmin(x_c, c_c)
34  delta_c <- ifelse(t_c == c_c, 0, 1)
35  tau_c <- c()
36  i <- 1
37  while (i <= count_c) {
38    sum_censored_c <- 0
39    censored_count_c <- 0
40    while (i <= count_c && delta_c[i] == 0) {
41      sum_censored_c <- sum_censored_c + t_c[i]
42      i <- i + 1
43    }
44    if (i <= count_c && delta_c[i] == 1) {
45      sum_censored_c <- sum_censored_c + t_c[i]
46    }
47    tau_c <- c(tau_c, sum_censored_c)
48    i <- i + 1
49  }
50
51
52  # Creiamo il processo di osservazione del gruppo sperimentale
53  x_e <- rexp(count_e, rate = lambda_e)
54  c_e <- rexp(count_e, rate = 0.008)
55  t_e <- pmin(x_e, c_e)
56  delta_e <- ifelse(t_e == c_e, 0, 1)
57  tau_e <- c()
58  i <- 1
59  while (i <= count_e) {
60    sum_censored_e <- 0
61    censored_count_e <- 0
62    while (i <= count_e && delta_e[i] == 0) {
63      sum_censored_e <- sum_censored_e + t_e[i]
64      i <- i + 1
65    }
66    if (i <= count_e && delta_e[i] == 1) {
67      sum_censored_e <- sum_censored_e + t_e[i]
68    }
69    tau_e <- c(tau_e, sum_censored_e)
70    i <- i + 1
71  }
72
73
74  # Ricaviamo il numero di arrivi di controllo ed esperimento
```

```
75    y_c <- length(tau_c)
76    y_e <- length(tau_e)
77
78    # Sommiamo e ricaviamo il numero di arrivi totali
79    y <- y_c + y_e
80
81    # Ricaviamo i tempi di osservazione di esperimento e controllo
82    E_c <- sum(tau_c)
83    E_e <- sum(tau_e)
84
85    # Calcolo del p-value
86    pvalue_list[h] <- 1 - ppois(y, lambda = (E_c + E_e) * lambda_c)
87  }
88
89  # Stampiamo i risultati delle simulazioni
90  print(pvalue_list)
91
92  # Contiamo quante volte il p-value    maggiore della soglia
93  count_greater_than_treshold <- sum(pvalue_list > treshold)
94  count_less_or_equal_treshold <- sum(pvalue_list <= treshold)
95
96  # Stampiamo i risultati
97  print(paste("p-value maggiore della soglia: ", count_greater_than_
        treshold))
98  print(paste("p-value minore della soglia: ", count_less_or_equal_
        treshold))
```

Capitolo 6

Confronto tra metodi e conclusioni

Per valutare i due modelli possiamo calcolare le stime dell'accuracy. Altre metriche come la precision o la recall, avendo regolato la soglia caso per caso, non avrebbero senso. Ricordiamo che

$$\widehat{accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Contando gli esiti della classificazione per tutti i casi studiati, si ottiene

$$\widehat{accuracy}_{bayes} = \frac{7077}{8000} = 0{,}885\%$$

$$\widehat{accuracy}_{freq} = \frac{7149}{8000} = 0{,}894\%$$

Ovviamente il comportamento ha una componente di aleatorietà. Ne segue che, differendo i risultati di cosi poco, si può affermare che i due metodi performino alla stessa maniera.

6.1 Comportamenti identici

Quindi le prestazioni del nuovo metodo sono perfettamente in linea con le prestazioni del BDRIBS. Quello che sorpende è che i comportamenti dei due metodi siano praticamente identici nei vari casi studiati. In particolare, per entrambi i modelli abbiamo che aumentando il numero di pazienti il modello miglora significativamente solo quando in input abbiamo un'effettiva "positività", e quindi la soglia viene regolata di conseguenza.
Anche la risposta rispetto alla variazione di k, che approssimativamente rappresenra il rapporto tra numero di pazienti di trattamento sperimentale e numero di

pazienti di controllo, è identica. Aumenteranno all'aumentare di k le prestazioni per entrambi i metodi, e le motivazioni sono quelle spiegate precedentemente. Inoltre quando $k > 1$, avendo più pazienti appartenenti al trattamento sperimentale, la rilevazione della "positività" diventa più semplice; si può quindi pensare di rendere più restrittive le condizioni che portano al lancio dell'allarme (aumentando la soglia per il metodo bayesiano e diminuendola per il frequentista). Per $k < 1$ succede l'opposto, avendo meno pazienti del trattamento sperimentale sarà più difficile rilevare che il nuovo trattamento è effettivamente più pericoloso, e quindi vanno favorite le condizioni che portano al lancio dell'allarme.
Quello che si è visto è quindi che, al migliorare delle condizioni in input (ovvero un numero di pazienti totali elevato e un rapporto k più alto), i due metodi migliorano sopratutto nel rilevare la "positività"; regolando la soglia al netto di questo comportamento, i due modelli migliorano anche nell'identificare correttamente i casi di "negatività".

6.2 Dallo stocastico al deterministico

Adesso vengono analizzati i due modelli sotto un punto di vista deterministico. Nel senso che non viene più effettuata alcuna simulazione, ma vengono dati in input ai due modelli tutti i valori di y di un determinato range, e ne viene calcolato il valore chiave del modello, ovvero la probabilità a posteriori che r sia maggiore di 1 nel caso bayesiano, e il p-value nel caso frequentista.
In figura 6.1 e 6.2 sono mostrati i risultati (a sinistra modello bayesiano a destra modello frequentista).

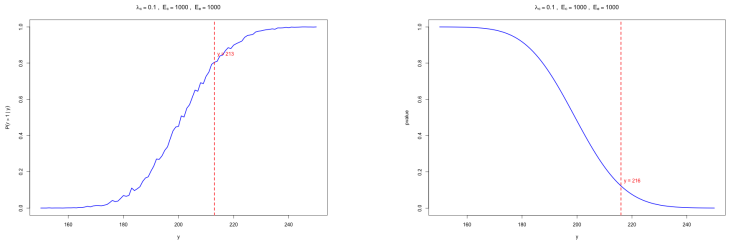

Figura 6.1

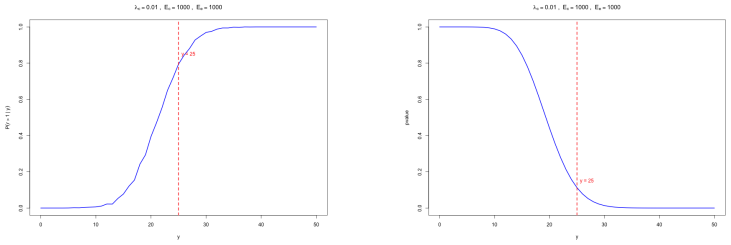

Figura 6.2

I valori a destra della linea verticale rossa, che è stata tirata in corrispondenza del punto di intersezione tra la curva ed i valori di soglia utilizzati precedentemente(0,8 per il primo e 0,12 per il secondo), sono i valori di y per il quale il modello lancia l'allarme.
L'andamento delle curve è crescente per il bayesiano e decrescente per il frequentista, e infatti nel primo caso la classificazione è positiva quando il valore è maggiore della soglia, nel secondo caso quando il valore è minore della soglia. Tuttavia mentre la curva del p-value è strettamente decrescente, si può notare come la curva della probabilità di r a posteriori, pur avendo un andamento crescente, in alcuni punti si abbassa leggermente per poi risalire. Questo comportamento è errato dal punto di vista logico, perchè la probabilità che r sia maggiore di uno dovrebbe sempre crescere all' aumentare di y.
La libreria Jags, utilizzata nei codici implementati per simulare il metodo bayesiano, utilizza metodi Markov chain Monte Carlo per generare campioni della distribuzione a posteriori. Quindi anche se non simuliamo il valore di y, l'output del modello sarà comunque aleatorio, e questo spiega perchè succede che all'aumentare di y la probabilità a posteriori che r sia maggiore di uno possa decrescere.

6.3 Conclusioni

Questa tesi ha esplorato un metodo statistico avanzato, il Bayesian Detection of potential Risk using Inference on Blinded Safety data (BDRIBS). In particolare è stato inizialmente spiegato il suo scopo, ovvero quello di indagare sui possibili rischi maggiori di un trattamento sperimentale rispetto a quello standard quando i dati a disposizione sono limitati. Successivamente è stata analizzata la struttura del modello, con particolare attenzione a quelle che sono le ipotesi del modello, e quali distibuzioni a priori utilizzare al variare di queste. Infine il metodo è stato simulato, variando più volte i parametri in input, e i risultati sono stati mostrati su bar plot per ottenere una maggiore interpretabilità.
Per far si che la simulazione si avvicinasse quanto più possibile a quello che avviene

nella realtà, sono stati costruiti i processi di osservazione in maniera rigorosa e coerente con ciò che accade negli studi clinici. L'unico elemento di rottura rispetto ai casi reali è rappresentato dal fatto che questi processi sono stati simulati separatamente per braccio di controllo e braccio sperimentale. Infatti per garantire che i tempi intercorrenti degli arrivi osservati siano distribuiti come un'esponenziale di parametro uguale al rate d'infezione, è necessario che il rate d'infezione sia univoco, il che è vero sotto l'ipotesi nulla, ma non è vero in generale. Un possibile miglioramento futuro potrebbe essere quello di simulare il metodo con un unico processo di osservazione, capire come varierebbe il rate dei tempi intercorrenti, e adeguare il modello a questa situazione.
Uno degli obiettivi di questo studio è stato quello di trovare, per ogni configurazione di parametri, quale soglia permettesse di raggiungere un equilibrio tra l'efficacia del metodo nel caso in cui il rate del nuovo trattamento sia effettivamente più pericoloso rispetto al trattamento standard, e l'efficacia del metodo quando i due trattamenti abbiano lo stesso rate d'infezione. Tuttavia, questo studio ha analizzato il comportamento del metodo solo confrontando i casi in cui $r = 1$ con quelli in cui $r = 2$. Un possibile lavoro più ampio potrebbe essere quello di estendere l'analisi confrontando i casi negativi ($r = 1$) con altri valori di r diversi da 2. L'ipotesi è che, aumentando il valore di r, anche la soglia d'equilibrio tenda a salire. Infatti, un rischio relativo più alto indica una maggiore probabilità di esiti avversi nel trattamento sperimentale rispetto a quello tradizionale. In queste situazioni, il metodo dovrebbe riuscire più facilmente a riconoscere correttamente i casi positivi, migliorando così le sue prestazioni. Al contrario, se il valore di r nei casi positivi è inferiore a 2, ci si aspetta una soglia d'equilibrio più bassa e una riduzione delle prestazioni del modello.
La seconda parte dell'articolo introduce un nuovo metodo, che abbia gli stessi obiettivi del BDRIBS, basato però su un approccio frequentista. Anche in questo caso è stata analizzata la struttura del modello, e successivamente sono state eseguite delle simulazioni, utilizzando gli stessi parametri di input che vengono utilizzati per simulare il modello bayesiano. I risultati hanno mostrato che questo metodo mantiene lo stesso livello di affidabilità, e che replica esattamente gli stessi comportamenti del BDRIBS al variare dei parametri in input.
Questo approccio però offre anche dei vantagi rispetto al metodo tradizionale. Innanzitutto la sua struttura è più corta e facilmente interpetabile; questa sua maggiore semplicità si riflette anche in termini computazionali, infatti il codice per simularlo è più breve e viene eseguito in molti meno secondi (circa 2 contro i 30 del metodo bayesiano). Inoltre, come visto nei capitoli precedenti, il suo output dipende esclusivamente dall'input, al contrario dell'altro metodo, il cui output ha una componente casuale. Questo è sicuramente apprezzabile e garantisce una maggiore robustezza.
In conclusione possiamo affermare che questo nuovo metodo frequentista rappresenti

una valida alternativa al tradizionale Bayesian Detection of potential Risk using Inference on Blinded Safety data.

Bibliografia

[1] Saurabh Mukhopadhyay, Brian Waterhouse, *Bayesian Detection of Potential Safety Signal from Blinded Clinical Trial Data*, presentation at DIA Bayesian Scientific Working Group KOL Lecture Series, AbbVie, June 2021.

[2] Brian Waterhouse, Alan Hartford, Saurabh Mukhopadhyay, Ryan Ferguson, Barbara A. Hendrickson. *Using the Bayesian detection of potential risk using inference on blinded safety data (BDRIBS) method to support the decision to refer an event for unblinded evaluation.* Pharmaceutical Statistics, 2022.