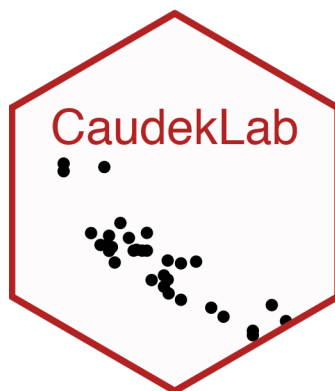


Psicometria

Corrado Caudek

Questo documento è stato realizzato con:

- \LaTeX e la classe memoir (<http://www.ctan.org/pkg/memoir>);
- R (<http://www.r-project.org/>) e RStudio (<http://www.rstudio.com/>);
- bookdown (<http://bookdown.org/>) e memoir (<https://ericmarcon.github.io/memoiR/>).



Nel blog della mia pagina personale sono forniti alcuni approfondimenti degli argomenti qui trattati.

<https://ccaudek.github.io/caudeklab/>

Indice

Indice	iii
Prefazione	v
La psicologia e la Data Science	v
Come studiare	vi
Sviluppare un metodo di studio efficace	vii
1 Modello Beta-Binomiale	1
1.1 Una proporzione	1
Il presidente Trump e l'idrossiclorochina	1
Interfaccia cmdstanr	2
Fase 1	2
Fase 2	4
Burn-in	5
Inferenza	5
La critica di Hulme et al. (2020)	9
1.2 Due proporzioni	9
Considerazioni conclusive	11
2 Diagnostica delle catene markoviane	13
2.1 Esame dei <i>trace plot</i>	13
2.2 Confronto delle catene parallele	15
2.3 Numerosità campionaria effettiva	18
2.4 Autocorrelazione	18
2.5 Statistica \hat{R}	21
2.6 Diagnostica di convergenza di Geweke	22
3 Sintesi a posteriori	25
3.1 Stima puntuale	25
3.2 Intervallo di credibilità	26
Intervallo di credibilità a code uguali	26
Intervallo di credibilità a densità a posteriori più alta	27
Interpretazione	27
3.3 Un esempio concreto	27
Stime puntuali della distribuzione a posteriori	28
Intervallo di credibilità	29
Probabilità della distribuzione a posteriori	30
Considerazioni conclusive	33

Bibliografia	35
Elenco delle figure	37

Prefazione

Data Science per psicologi contiene il materiale delle lezioni dell'insegnamento di *Psicometria B000286* (A.A. 2021/2022) rivolto agli studenti del primo anno del Corso di Laurea in Scienze e Tecniche Psicologiche dell'Università degli Studi di Firenze.

L'insegnamento di Psicometria si propone di fornire agli studenti un'introduzione all'analisi dei dati in psicologia. Le conoscenze/competenze che verranno sviluppate in questo insegnamento sono quelle della *Data science*, ovvero le conoscenze/competenze che si pongono all'intersezione tra statistica (ovvero, richiedono la capacità di comprendere teoremi statistici) e informatica (ovvero, richiedono la capacità di sapere utilizzare un software).

La psicologia e la Data Science

It's worth noting, before getting started, that this material is hard. If you find yourself confused at any point, you are normal. Any sense of confusion you feel is just your brain correctly calibrating to the subject matter. Over time, confusion is replaced by comprehension [...] — Richard McElreath

Sembra sensato spendere due parole su un tema che è importante per gli studenti: quello indicato dal titolo di questo Capitolo. È ovvio che agli studenti di psicologia la statistica non piace. Se piacesse, forse studierebbero Data Science e non psicologia; ma non lo fanno. Di conseguenza, gli studenti di psicologia si chiedono: “perché dobbiamo perdere tanto tempo a studiare queste cose quando in realtà quello che ci interessa è tutt'altro?” Questa è una bella domanda.

C'è una ragione molto semplice che dovrebbe farci capire perché la Data Science è così importante per la psicologia. Infatti, a ben pensarci, la psicologia è una disciplina intrinsecamente statistica, se per statistica intendiamo quella disciplina che studia la variazione delle caratteristiche degli individui nella popolazione. La psicologia studia *gli individui* ed è proprio la variabilità inter- e intra-individuale ciò che vogliamo descrivere e, in certi casi, predire. In questo senso, la psicologia è molto diversa dall'ingegneria, per esempio. Le proprietà di un determinato ponte sotto certe condizioni, ad esempio, sono molto simili a quelle di un altro ponte, sotto le medesime condizioni. Quindi, per un ingegnere la statistica è poco importante: le proprietà dei materiali sono unicamente dipendenti dalla loro composizione e restano costanti. Ma lo stesso non può dirsi degli individui: ogni individuo è unico e cambia nel tempo. E le variazioni tra gli individui, e di un individuo nel tempo, sono l'oggetto di studio proprio della

psicologia: è dunque chiaro che i problemi che la psicologia si pone sono molto diversi da quelli affrontati, per esempio, dagli ingegneri. Questa è la ragione per cui abbiamo tanto bisogno della *data science* in psicologia: perché la *data science* ci consente di descrivere la variazione e il cambiamento. E queste sono appunto le caratteristiche di base dei fenomeni psicologici.

Sono sicuro che, leggendo queste righe, a molti studenti sarà venuta in mente la seguente domanda: perché non chiediamo a qualche esperto di fare il “lavoro sporco” (ovvero le analisi statistiche) per noi, mentre noi (gli psicologi) ci occupiamo solo di ciò che ci interessa, ovvero dei problemi psicologici slegati dai dettagli “tecnici” della *data science*? La risposta a questa domanda è che non è possibile progettare uno studio psicologico sensato senza avere almeno una comprensione rudimentale della *data science*. Le tematiche della *data science* non possono essere ignorate né dai ricercatori in psicologia né da coloro che svolgono la professione di psicologo al di fuori dell’Università. Infatti, anche i professionisti al di fuori dall’università non possono fare a meno di leggere la letteratura psicologica più recente: il continuo aggiornamento delle conoscenze è infatti richiesto dalla deontologia della professione. Ma per potere fare questo è necessario conoscere un bel po’ di *data science*! Basta aprire a caso una rivista specialistica di psicologia per rendersi conto di quanto ciò sia vero: gli articoli che riportano i risultati delle ricerche psicologiche sono zeppi di analisi statistiche e di modelli formali. E la comprensione della letteratura psicologica rappresenta un requisito minimo nel bagaglio professionale dello psicologo.

Le considerazioni precedenti cercano di chiarire il seguente punto: la *data science* non è qualcosa da studiare a malincuore, in un singolo insegnamento universitario, per poi poterla tranquillamente dimenticare. Nel bene e nel male, gli psicologi usano gli strumenti della *data science* in tantissimi ambiti della loro attività professionale: in particolare quando costruiscono, somministrano e interpretano i test psicometrici. È dunque chiaro che possedere delle solide basi di *data science* è un tassello imprescindibile del bagaglio professionale dello psicologo. In questo insegnamento verranno trattati i temi base della *data science* e verrà adottato un punto di vista bayesiano, che corrisponde all’approccio più recente e sempre più diffuso in psicologia.

Come studiare

I know quite certainly that I myself have no special talent. Curiosity, obsession and dogged endurance, combined with self-criticism, have brought me to my ideas. — Albert Einstein

Il giusto metodo di studio per prepararsi all’esame di Psicometria è quello di seguire attivamente le lezioni, assimilare i concetti via via che essi vengono presentati e verificare in autonomia le procedure presentate a lezione. Incoraggio gli studenti a farmi domande per chiarire ciò che non è stato capito appieno. Incoraggio gli studenti a utilizzare i forum attivi su Moodle e, soprattutto, a svolgere gli esercizi proposti su Moodle. I problemi forniti su Moodle rappresentano il livello di difficoltà richiesto per superare l’esame e consentono allo studente di comprendere se le competenze sviluppate fino a quel punto sono sufficienti rispetto alle richieste dell’esame.

La prima fase dello studio, che è sicuramente individuale, è quella in cui è necessario acquisire le conoscenze teoriche relative ai problemi che saranno presentati all'esame. La seconda fase di studio, che può essere facilitata da scambi con altri e da incontri di gruppo, porta ad acquisire la capacità di applicare le conoscenze: è necessario capire come usare un software (R) per applicare i concetti statistici alla specifica situazione del problema che si vuole risolvere. Le due fasi non sono però separate: il saper fare molto spesso ci aiuta a capire meglio.

Sviluppare un metodo di studio efficace

Memorization is not learning. — Richard Phillips Feynman

Avendo insegnato Psicometria molte volte in passato ho notato nel corso degli anni che gli studenti con l'atteggiamento mentale che descriverò qui sotto generalmente ottengono ottimi risultati. Alcuni studenti sviluppano naturalmente questo approccio allo studio, ma altri hanno bisogno di fare uno sforzo per maturarlo. Fornisco qui sotto una breve descrizione del "metodo di studio" che, nella mia esperienza, è il più efficace per affrontare le richieste di questo insegnamento.

- Dedicate un tempo sufficiente al materiale di base, apparentemente facile; assicuratevi di averlo capito bene. Cercate le lacune nella vostra comprensione. Leggere presentazioni diverse dello stesso materiale (in libri o articoli diversi) può fornire nuove intuizioni.
- Gli errori che facciamo sono i nostri migliori maestri. Istitivamente cerchiamo di dimenticare subito i nostri errori. Ma il miglior modo di imparare è apprendere dagli errori che commettiamo. In questo senso, una soluzione corretta è meno utile di una soluzione sbagliata. Quando commettiamo un errore questo ci fornisce un'informazione importante: ci fa capire qual è il materiale di studio sul quale dobbiamo ritornare e che dobbiamo capire meglio.
- C'è ovviamente un aspetto "psicologico" nello studio. Quando un esercizio o problema ci sembra incomprensibile, la cosa migliore da fare è dire: "mi arrendo", "non ho idea di cosa fare!". Questo ci rilassa: ci siamo già arresi, quindi non abbiamo niente da perdere, non dobbiamo più preoccuparci. Ma non dobbiamo fermarci qui. Le cose "migliori" che faccio (se ci sono) le faccio quando non ho voglia di lavorare. Alle volte, quando c'è qualcosa che non so fare e non ho idea di come affrontare, mi dico: "oggi non ho proprio voglia di fare fatica", non ho voglia di mettermi nello stato mentale per cui "in 10 minuti devo risolvere il problema perché dopo devo fare altre cose". Però ho voglia di *divertirmi* con quel problema e allora mi dedico a qualche aspetto "marginale" del problema, che so come affrontare, oppure considero l'aspetto più difficile del problema, quello che non so come risolvere, ma invece di cercare di risolverlo, guardo come altre persone hanno affrontato problemi simili, oppure lo stesso problema in un altro contesto. Non mi pongo l'obiettivo "risolvi il problema in 10 minuti", ma invece

quello di farmi un'idea "generale" del problema, o quello di capire un caso più specifico e più semplice del problema. Senza nessuna pressione. Infatti, in quel momento ho deciso di non lavorare (ovvero, di non fare fatica). Va benissimo se "parto per la tangente", ovvero se mi metto a leggere del materiale che sembra avere poco a che fare con il problema centrale (le nostre intuizioni e la nostra curiosità solitamente ci indirizzano sulla strada giusta). Quando faccio così, molto spesso trovo la soluzione del problema che mi ero posto e, paradossalmente, la trovo in un tempo minore di quello che, in precedenza, avevo dedicato a "lavorare" al problema. Allora perché non faccio sempre così? C'è ovviamente l'aspetto dei "10 minuti" che non è sempre facile da dimenticare. Sotto pressione, possiamo solo agire in maniera automatica, ovvero possiamo solo applicare qualcosa che già sappiamo fare. Ma se dobbiamo imparare qualcosa di nuovo, la pressione è un impedimento.

- È utile farsi da soli delle domande sugli argomenti trattati, senza limitarsi a cercare di risolvere gli esercizi che vengono assegnati. Quando studio qualcosa mi viene in mente: "se questo è vero, allora deve succedere quest'altra cosa". Allora verifico se questo è vero, di solito con una simulazione. Se i risultati della simulazione sono quelli che mi aspetto, allora vuol dire che ho capito. Se i risultati sono diversi da quelli che mi aspettavo, allora mi rendo conto di non avere capito e ritorno indietro a studiare con più attenzione la teoria che pensavo di avere capito – e ovviamente mi rendo conto che c'era un aspetto che avevo frainteso. Questo tipo di verifica è qualcosa che dobbiamo fare da soli, in prima persona: nessun altro può fare questo al posto nostro.
- Non aspettatevi di capire tutto la prima volta che incontrate un argomento nuovo. È utile farsi una nota mentalmente delle lacune nella vostra comprensione e tornare su di esse in seguito per cercare di colmarle. L'atteggiamento naturale, quando non capiamo i dettagli di qualcosa, è quello di pensare: "non importa, ho capito in maniera approssimativa questo punto, non devo preoccuparmi del resto". Ma in realtà non è vero: se la nostra comprensione è superficiale, quando il problema verrà presentato in una nuova forma, non riusciremo a risolverlo. Per cui i dubbi che ci vengono quando studiamo qualcosa sono il nostro alleato più prezioso: ci dicono esattamente quali sono gli aspetti che dobbiamo approfondire per potere migliorare la nostra preparazione.
- È utile sviluppare una visione d'insieme degli argomenti trattati, capire l'obiettivo generale che si vuole raggiungere e avere chiaro il contributo che i vari pezzi di informazione forniscono al raggiungimento di tale obiettivo. Questa organizzazione mentale del materiale di studio facilita la comprensione. È estremamente utile creare degli schemi di ciò che si sta studiando. Non aspettate che sia io a fornirvi un riepilogo di ciò che dovete imparare: sviluppate da soli tali schemi e tali riassunti.
- Tutti noi dobbiamo imparare l'arte di trovare le informazioni, non solo nel caso di questo insegnamento. Quando vi trovate di fronte a qualcosa

che non capite, o ottenete un oscuro messaggio di errore da un software,
ricordatevi: “Google is your friend”.

Corrado Caudek

Febbraio 2022

Modello Beta-Binomiale

1.1 Una proporzione

Si considerino n variabili casuali Bernoulliane i.i.d.:

$$y = (y_1, \dots, y_n) \stackrel{iid}{\sim} \mathcal{B}(\theta).$$

Vogliamo stimare θ avendo osservato y . Essendo i.i.d., i dati possono essere riassunti dal numero totale di successi nelle n prove, denotato da y . Il modello binomiale è

$$p(y | \theta) = \text{Bin}(y | n, \theta) = \binom{n}{y} \theta^y (1 - \theta)^{n-y}, \quad (1.1)$$

dove nel termine di sinistra dell'equazione abbiamo ignorato n in quanto viene considerato fisso per disegno.

L'inferenza sul modello binomiale richiede di assegnare una distribuzione a priori su θ che dipende dall'informazione disponibile a priori. Se scegliamo, ad esempio, una $B(2, 2)$ quale distribuzione a priori, il modello diventa:

$$\begin{aligned} y &\sim \text{Bin}(n, \theta) \\ \theta &\sim B(2, 2), \end{aligned} \quad (1.2)$$

dove la prima riga definisce la funzione di verosimiglianza e la seconda riga definisce la distribuzione a priori. Sulla base di ciò che è stato detto nel Capitolo ??, sappiamo che le equazioni (1.2) definiscono il caso Beta-Binomiale.

Il presidente Trump e l'idrossiclorochina

Per fare un esempio concreto, consideriamo un set di dati reali. Cito dal *Washington Post* del 7 aprile 2020:

One of the most bizarre and disturbing aspects of President Trump's nightly press briefings on the coronavirus pandemic is when he turns

into a drug salesman. Like a cable TV pitchman hawking ‘male enhancement’ pills, Trump regularly extols the virtues of taking hydroxychloroquine, a drug used to treat malaria and lupus, as a potential ‘game changer’ that just might cure Covid-19.

Tralasciamo qui il fatto che il presidente Trump non è un esperto in questo campo. Esaminiamo invece le evidenze iniziali a supporto dell’ipotesi che l’idrossiclorochina possa essere utile per la cura del Covid-19, ovvero le evidenze che erano disponibili nel momento in cui il presidente Trump ha fatto le affermazioni riportate sopra (in seguito, quest’idea è stata screditata). Tali evidenze sono state fornite da uno studio di Gautret et al. (2020). Il disegno sperimentale di Gautret et al. (*ibid.*) comprende, tra le altre cose, il confronto tra una condizione sperimentale e una condizione di controllo. Il confronto importante è tra la proporzione di paziente positivi al virus SARS-CoV-2 nel gruppo sperimentale (a cui è stata somministrata l’idrossiclorochina; 6 su 14) e la proporzione di paziente positivi nel gruppo di controllo (a cui non è stata somministrata l’idrossiclorochina; ovvero 14 su 16). Obiettivo di questo Capitolo è mostrare come si possa fare inferenza sul modello (1.2) usando il linguaggio Stan.

Interfaccia `cmdstanr`

Nella seguente discussione verrà ottenuta una stima bayesiana del parametro θ usando l’interfaccia `cmdstanr` di `CmdStan`.¹ Considereremo qui solo il gruppo di controllo. Iniziamo a caricare i pacchetti necessari:

```
library("cmdstanr")
set_cmdstan_path("/Users/corrado/.cmdstan/cmdstan-2.28.0")
library("posterior")
rstan_options(auto_write = TRUE) # avoid recompilation of models
options(mc.cores = parallel::detectCores()) # parallelize across all CPUs
Sys.setenv(LOCAL_CPPFLAGS = "-march=native") # improve execution time
SEED <- 374237 # set random seed for reproducibility
```

Ci sono due passaggi essenziali per le analisi svolte mediante `cmdstanr`:

1. definire la struttura del modello bayesiano nella notazione Stan;
2. eseguire il campionamento della distribuzione a posteriori.

Esaminiamo questi due passaggi nel contesto del modello Beta-Binomiale definito dalla (1.2).

Fase 1

È necessario definire i dati, i parametri e il modello. I *dati* del gruppo di controllo, che verrà qui esaminato, devono essere contenuti in un oggetto di classe `list`:

¹I modelli discussi in questo capitolo sono discussi da Gelman et al. (1995) mentre il codice è stato ricavato dalla seguente [pagina web](#).

```
data1_list <- list(  
  N = 16,  
  y = c(rep(1, 14), rep(0, 2))  
)
```

Il modello dipende dal *parametro* θ . In Stan, dobbiamo specificare che θ può essere un qualsiasi numero reale compreso tra 0 e 1.

Il *modello* è $\text{Bin}(n, \theta)$ e, nel linguaggio Stan, può essere scritto come

```
for (i in 1:N) {  
  y[i] ~ bernoulli(theta)  
}
```

ovvero come

```
y ~ bernoulli(theta)
```

La struttura del modello Beta-Binomiale viene tradotta nella sintassi Stan² e viene poi memorizzata come stringa di caratteri del file `oneprop1.stan`:

```
modelString <- "  
data {  
  int<lower=0> N;  
  int<lower=0, upper=1> y[N];  
}  
parameters {  
  real<lower=0, upper=1> theta;  
}  
model {  
  theta ~ beta(2, 2);  
  y ~ bernoulli(theta);  
  // the notation using ~ is syntactic sugar for  
  // target += beta_lpdf(theta | 1, 1); // lpdf for continuous theta  
  // target += bernoulli_lpmf(y | theta); // lpmf for discrete y  
  // target is the log density to be sampled  
  //  
  // y is an array of integers and  
  // y ~ bernoulli(theta);  
  // is equivalent to  
  // for (i in 1:N) {  
  //   y[i] ~ bernoulli(theta);  
  // }  
  // which is equivalent to  
  // for (i in 1:N) {  
  //   target += bernoulli_lpmf(y[i] | theta);  
  // }
```

²Si veda l'Appendice ??

```
}  
generated quantities {  
  int y_rep[N];  
  real log_lik[N];  
  for (n in 1:N) {  
    y_rep[n] = bernoulli_rng(theta);  
    log_lik[n] = bernoulli_lpmf(y[n] | theta);  
  }  
}  
"  
writeLines(modelString, con = "code/oneprop1.stan")
```

Fase 2

Leggiamo l'indirizzo del file che contiene il codice Stan:

```
file <- file.path("code", "oneprop1.stan")
```

Compiliamo il codice:

```
mod <- cmdstan_model(file)
```

Il campionamento MCMC si realizza con la chiamata:

```
fit1 <- mod$sample(  
  data = data1_list,  
  iter_sampling = 4000L,  
  iter_warmup = 2000L,  
  seed = SEED,  
  chains = 4L,  
  parallel_chains = 4L,  
  refresh = 0,  
  thin = 1  
)
```

Avendo assunto una distribuzione a priori per il parametro θ , l'algoritmo procede in maniera ciclica, correggendo la distribuzione a priori di θ condizionandola ai valori già generati. Dopo un certo numero di cicli, necessari per portare l'algoritmo a convergenza, i valori estratti possono essere assunti come campionati dalla distribuzione a posteriori di θ .

Si noti che `$sample()` richiede due tipi di informazioni. Innanzitutto, dobbiamo specificare le informazioni sul modello in base a:

- `mod` = la stringa di caratteri che definisce il modello (qui `oneprop1.stan`),
- `data` = i dati in formato lista (`data1_list`).

Dobbiamo inoltre specificare le informazioni sul campionamento MCMC utilizzando tre argomenti aggiuntivi:

- L'argomento `chains` specifica quante catene di Markov parallele eseguire. Eseguiamo qui quattro catene, quindi otteniamo quattro campioni distinti di valori π .
- L'argomento `iter` specifica il numero desiderato di iterazioni o la lunghezza di ciascuna catena di Markov. Per impostazione predefinita, la prima metà di queste iterazioni è costituita da campioni “burn-in” o “warm-up” che verranno ignorati. La seconda metà è conservata e costituisce un campione della distribuzione a posteriori.
- L'argomento `seed` per impostare il numero casuale che genera il seme per una simulazione `cmdstanr`.

Burn-in

Al crescere del numero di passi della catena, la distribuzione di target viene sempre meglio approssimata. All'inizio del campionamento, però, la distribuzione può essere significativamente lontana da quella stazionaria, e ci vuole un certo tempo prima di raggiungere la distribuzione stazionaria di equilibrio, detto, appunto, periodo di *burn-in*. I campioni provenienti da tale parte iniziale della catena vanno tipicamente scartati perché possono non rappresentare accuratamente la distribuzione a posteriori

Inferenza

Un sommario della distribuzione a posteriori si ottiene con:

```
fit1$summary(c("theta"))
#> # A tibble: 1 x 10
#>   variable mean median    sd    mad    q5   q95  rhat
#>   <chr>    <dbl>  <dbl>  <dbl>  <dbl> <dbl> <dbl> <dbl>
#> 1 theta    0.802  0.813 0.0868 0.0867 0.644 0.928 1.00
#> # ... with 2 more variables: ess_bulk <dbl>, ess_tail <dbl>
```

Creiamo un oggetto di classe `stanfit`

```
stanfit1 <- rstan::read_stan_csv(fit1$output_files())
```

di dimensioni

```
dim(as.matrix(stanfit1, pars = "theta"))
#> [1] 16000    1
```

I primi 10 valori sono presentati qui di seguito

```
as.matrix(stanfit1, pars = "theta") %>%
  head(10)
#>           parameters
#> iterations    theta
```

```
#>      [1,] 0.757239
#>      [2,] 0.703550
#>      [3,] 0.772169
#>      [4,] 0.747881
#>      [5,] 0.765079
#>      [6,] 0.793709
#>      [7,] 0.857447
#>      [8,] 0.845012
#>      [9,] 0.824972
#>     [10,] 0.881219
```

La matrice precedente include i valori assunti dalla catena di Markov, ovvero un insieme di valori plausibili θ estratti dalla distribuzione a posteriori.

Un tracciato della catena di Markov illustra questa esplorazione rappresentando il valore θ sulle ordinate e l'indice progressivo di in ogni iterazione sull'ascissa. Usiamo la funzione `mcmc_trace()` del pacchetto `bayesplot` (Gabry et al. 2019) per costruire il grafico che include tutte e quattro le catene di Markov:

```
stanfit1 %>%
  mcmc_trace(pars = c("theta"), size = 0.1)
```

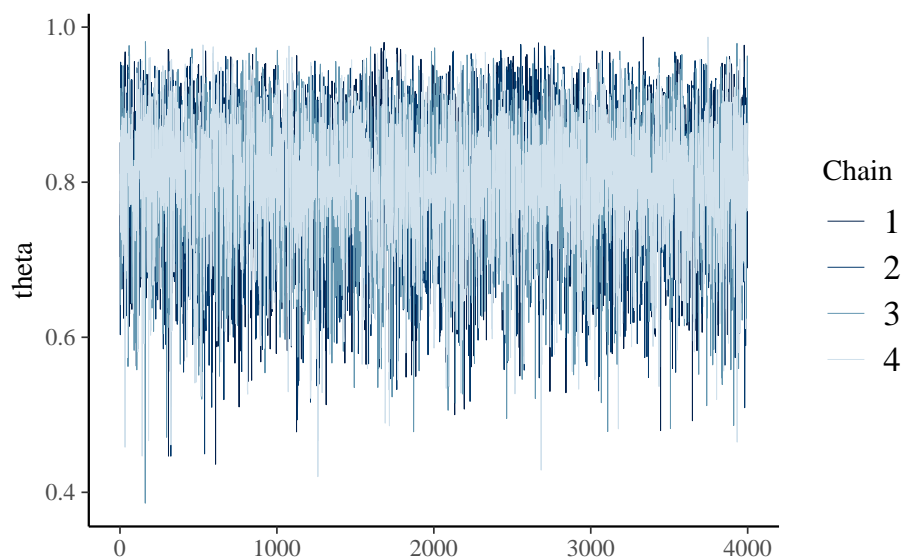


Figura 1.1: Trace-plot per il parametro θ nel modello Beta-Binomiale.

La figura 1.1 mostra che le catene esplorano uno spazio compreso approssimativamente tra 0.7 e 0.9; tale figura descrive il comportamento *longitudinale* delle catene di Markov.

Possiamo anche esaminare la distribuzione degli stati della catena di Markov, ovvero, dei valori che queste catene visitano lungo il loro percorso, ignorando l'ordine di queste visite. L'istogramma della figura 1.2 fornisce una rappresentazione grafica di questa distribuzione per i 16000 valori complessivi delle quattro catene, ovvero per 4000 valori provenienti da ciascuna catena.


```
mcmc_hist(stanfit1, pars = "theta") +
  yaxis_text(TRUE) +
  ylab("count")
```

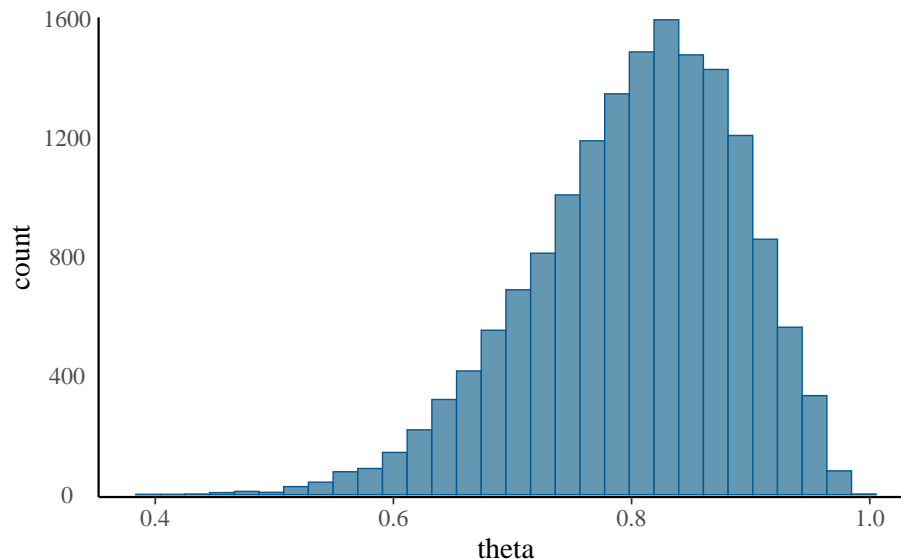


Figura 1.2: Istogramma che illustra l'approssimazione della distribuzione a posteriori per il parametro θ nel modello Beta-Binomiale.

Nel modello Beta-Binomiale in cui la verosimiglianza è binomiale con 14 successi su 16 prove e in cui assumiamo una distribuzione a priori di tipo Beta(2, 2) sul parametro θ , la distribuzione a posteriori è ancora una distribuzione Beta di parametri $\alpha = 2 + 14$ e $\beta = 2 + 16 - 14$. La figura 1.3 riporta un kernel density plot per i valori delle quattro catene di Markov con sovrapposta in nero la densità Beta(16, 4). Il punto importante è che la distribuzione dei valori delle catene di Markov produce un'eccellente approssimazione alla distribuzione bersaglio.³

```
mcmc_dens(stanfit1, pars = "theta") +
  yaxis_text(TRUE) +
  ylab("density") +
  stat_function(fun = dbeta, args = list(shape1 = 16, shape2 = 4))
```

Un intervallo di credibilità al 95% per θ si ottiene con la seguente chiamata:

```
posterior1 <- extract(stanfit1)
rstantools::posterior_interval(as.matrix(stanfit1), prob = 0.95)
#>           2.5%      97.5%
#> theta      0.6107860  0.94402990
#> y_rep[1]    0.0000000  1.00000000
```

³Nel caso presente, il risultato è poco utile dato che è disponibile una soluzione analitica. Tuttavia, questo esercizio mette in evidenza il fatto cruciale che, nei casi in cui possiamo verificarne la soluzione, il campionamento Monte Carlo a catena di Markov è in grado di trovare la risposta corretta. Di conseguenza, possiamo anche essere sicuri che fornirà un'approssimazione alla distribuzione a posteriori anche in quei casi in cui una soluzione analitica non è disponibile.

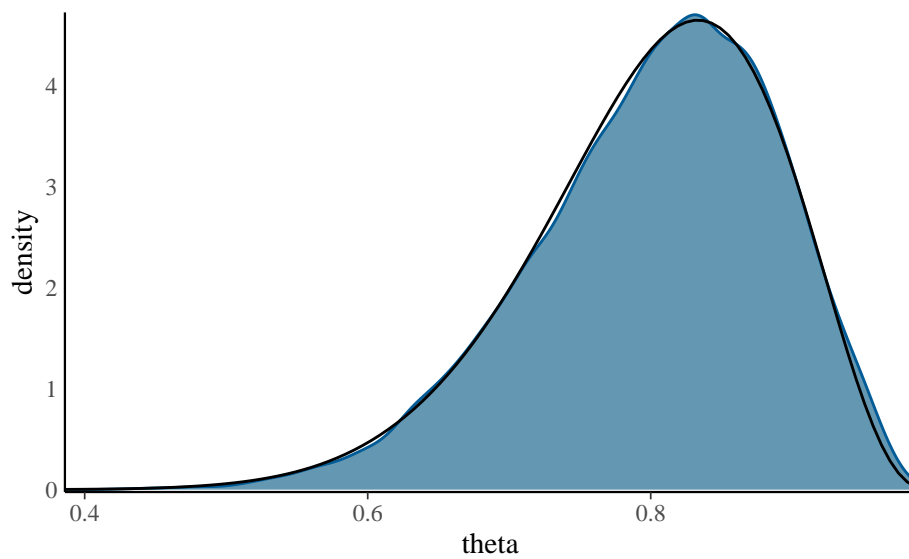


Figura 1.3: Istogramma che illustra l'approssimazione della distribuzione a posteriori per il parametro θ nel modello Beta-Binomiale. La curva nera rappresenta la corretta distribuzione a posteriori $\text{Beta}(16, 4)$.

```
#> y_rep[2]      0.0000000  1.0000000
#> y_rep[3]      0.0000000  1.0000000
#> y_rep[4]      0.0000000  1.0000000
#> y_rep[5]      0.0000000  1.0000000
#> y_rep[6]      0.0000000  1.0000000
#> y_rep[7]      0.0000000  1.0000000
#> y_rep[8]      0.0000000  1.0000000
#> y_rep[9]      0.0000000  1.0000000
#> y_rep[10]     0.0000000  1.0000000
#> y_rep[11]     0.0000000  1.0000000
#> y_rep[12]     0.0000000  1.0000000
#> y_rep[13]     0.0000000  1.0000000
#> y_rep[14]     0.0000000  1.0000000
#> y_rep[15]     0.0000000  1.0000000
#> y_rep[16]     0.0000000  1.0000000
#> log_lik[1]    -0.4930093 -0.05759735
#> log_lik[2]    -0.4930093 -0.05759735
#> log_lik[3]    -0.4930093 -0.05759735
#> log_lik[4]    -0.4930093 -0.05759735
#> log_lik[5]    -0.4930093 -0.05759735
#> log_lik[6]    -0.4930093 -0.05759735
#> log_lik[7]    -0.4930093 -0.05759735
#> log_lik[8]    -0.4930093 -0.05759735
#> log_lik[9]    -0.4930093 -0.05759735
#> log_lik[10]   -0.4930093 -0.05759735
#> log_lik[11]   -0.4930093 -0.05759735
#> log_lik[12]   -0.4930093 -0.05759735
#> log_lik[13]   -0.4930093 -0.05759735
```

```
#> log_lik[14] -0.4930093 -0.05759735
#> log_lik[15] -2.8829362 -0.94362543
#> log_lik[16] -2.8829362 -0.94362543
#> lp__        -12.7342100 -10.00860000
```

Svolgendo un’analisi bayesiana simile a questa, Gautret et al. (2020) hanno trovato che gli intervalli di credibilità del gruppo di controllo e del gruppo sperimentale non si sovrappongono. Questo fatto viene interpretato dicendo che il parametro θ è diverso nei due gruppi. Sulla base di queste evidenze, Gautret et al. (ibid.) hanno concluso, con un grado di certezza soggettiva del 95%, che nel gruppo sperimentale vi è una probabilità più bassa di risultare positivi al SARS-CoV-2 rispetto al gruppo di controllo. In altri termini, questa analisi dei dati suggerisce che l’idrossiclorochina sia efficace come terapia per il Covid-19.

La critica di Hulme et al. (2020)

Un articolo pubblicato da Hulme et al. (ibid.) si è posto il problema di rianalizzare i dati di Gautret et al. (2020).⁴ Tra gli autori di questo articolo figura anche Eric-Jan Wagenmakers, uno psicologo molto conosciuto per i suoi contributi metodologici. Hulme et al. (2020) hanno osservato che, nelle analisi statistiche riportate, Gautret et al. (2020) hanno escluso alcuni dati. Nel gruppo sperimentale, infatti, vi erano alcuni pazienti i quali, anziché migliorare, sono in realtà peggiorati. L’analisi statistica di Gautret et al. (ibid.) ha escluso i dati di questi pazienti. Se consideriamo tutti i pazienti — non solo quelli selezionati da Gautret et al. (ibid.) — la situazione diventa la seguente:

- gruppo sperimentale: 10 positivi su 18;
- gruppo di controllo: 14 positivi su 16.

L’analisi dei dati proposta da Hulme et al. (2020) richiede l’uso di alcuni strumenti statistici che, in queste dispense, non verranno discussi. Ma possiamo giungere alle stesse conclusioni raggiunte da questi ricercatori anche usando le procedure statistiche descritte nel Paragrafo successivo.

1.2 Due proporzioni

Svolgiamo ora l’analisi considerando tutti i dati, come suggerito da Hulme et al. (ibid.). Per fare questo verrà creato un modello bayesiano per fare inferenza sulla differenza tra due proporzioni. Una volta generate le distribuzioni a posteriori per le proporzioni di “successi” nei due gruppi, verrà anche generata la quantità

$$\omega = \frac{\theta_2/(1-\theta_2)}{\theta_1/(1-\theta_1)},$$

ovvero il rapporto tra gli Odds di positività tra i pazienti del gruppo di controllo e gli Odds di positività tra i pazienti del gruppo sperimentale. Se il valore dell’OR è uguale a 1, significa che l’Odds di positività nel gruppo di controllo

⁴Si veda <https://osf.io/5dgmX/>.

è uguale all'odds di positività nel gruppo sperimentale, cioè il fattore in esame (somministrazione dell'idrossiclorochina) è influente sulla comparsa della malattia. L'inferenza statistica sull'efficacia dell'idrossiclorochina come terapia per il Covid-19 può dunque essere effettuata esaminando l'intervallo di credibilità al 95% per l'OR: se tale intervallo include il valore 1, allora non vi è evidenza che l'idrossiclorochina sia efficace come terapia per il Covid-19.

Nell'implementazione di questo modello, la quantità di interesse è dunque l'odds ratio; tale quantità viene calcolata nel blocco `generated quantities` del programma Stan. In questo esempio useremo delle distribuzioni a priori vagamente informative per i parametri θ_1 e θ_2 .

```
data_list <- list(  
  N1 = 18,  
  y1 = 10,  
  N2 = 16,  
  y2 = 14  
)
```

```
modelString <- "  
// Comparison of two groups with Binomial  
data {  
  int<lower=0> N1;          // number of experiments in group 1  
  int<lower=0> y1;          // number of deaths in group 1  
  int<lower=0> N2;          // number of experiments in group 2  
  int<lower=0> y2;          // number of deaths in group 2  
}  
parameters {  
  real<lower=0,upper=1> theta1; // probability of death in group 1  
  real<lower=0,upper=1> theta2; // probability of death in group 2  
}  
model {  
  theta1 ~ beta(2, 2);      // prior  
  theta2 ~ beta(2, 2);      // prior  
  y1 ~ binomial(N1, theta1); // observation model / likelihood  
  y2 ~ binomial(N2, theta2); // observation model / likelihood  
}  
generated quantities {  
  // generated quantities are computed after sampling  
  real oddsratio = (theta2/(1-theta2))/(theta1/(1-theta1));  
}  
"  
writeLines(modelString, con = "code/twoprop1.stan")
```

```
file <- file.path("code", "twoprop1.stan")
```

```
mod <- cmdstan_model(file)
```

```
fit <- mod$sample(
  data = data_list,
  iter_sampling = 4000L,
  iter_warmup = 2000L,
  seed = SEED,
  chains = 4L,
  parallel_chains = 4L,
  refresh = 0,
  thin = 1
)
```

```
stanfit <- rstan::read_stan_csv(fit$output_files())
```

```
print(
  stanfit,
  pars = c("theta1", "theta2", "oddsratio"),
  digits_summary = 3L
)
#> Inference for Stan model: twoprop1-202110161322-1-604a8b.
#> 4 chains, each with iter=6000; warmup=2000; thin=1;
#> post-warmup draws per chain=4000, total post-warmup draws=16000.
#>
#>          mean se_mean   sd  2.5%  25%  50%  75%
#> theta1    0.546    0.001 0.104 0.337 0.475 0.547 0.619
#> theta2    0.801    0.001 0.087 0.605 0.747 0.812 0.865
#> oddsratio 4.859    0.049 4.740 0.914 2.221 3.599 5.933
#>
#>          97.5% n_eff  Rhat
#> theta1      0.743 11214 1.000
#> theta2      0.939 12359 1.000
#> oddsratio 16.251  9207 1.001
#>
#> Samples were drawn using NUTS(diag_e) at Sab Ott 16 13:22:40 2021.
#> For each parameter, n_eff is a crude measure of effective sample size,
#> and Rhat is the potential scale reduction factor on split chains (at
#> convergence, Rhat=1).
```

L'intervallo di credibilità del 95% per l'OR include il valore di 1.0 (ovvero, il valore che indica che gli odds di positività sono uguali nei due gruppi). In base agli standard correnti, un risultato di questo tipo non viene considerato come evidenza sufficiente per potere concludere che il parametro θ assume un valore diverso nei due gruppi. In altri termini, se consideriamo tutti i dati, e non solo quelli selezionati dagli autori della ricerca originaria, non vi è evidenza alcuna che l'idrossiclorochina sia efficace come terapia per il Covid-19.

Considerazioni conclusive

Concludiamo questa discussione dicendo che ciò che è stato presentato in questo capitolo è un esercizio didattico: la ricerca di Gautret et al. (2020) include

tante altre informazioni che non sono state qui considerate. Tuttavia, notiamo che la semplice analisi statistica che abbiamo qui descritto è stata in grado di replicare le conclusioni a cui sono giunti (per altra via) Hulme et al. ([2020](#)).

Diagnostica delle catene markoviane

Come discusso nel Paragrafo 1.1, le catene di Markov forniscono un'approssimazione che tende a convergere alla distribuzione a posteriori. “Approssimazione” e “convergenza” sono le parole chiave qui: il punto è che il campionamento MCMC non è perfetto. Questo solleva le seguenti domande:

- A cosa corrisponde, dal punto di vista grafico, una “buona” catena di Markov?
- Come possiamo sapere se il campione prodotto dalla catena di Markov costituisce un'approssimazione adeguata della distribuzione a posteriori?
- Quanto deve essere grande la dimensione del campione casuale prodotto dalla catena Markov?

Rispondere a queste ed altre domande di questo tipo fa parte di quell'insieme di pratiche che vano sotto il nome di *diagnostica delle catene Markoviane*.

La diagnostica delle catene Markoviane non è “una scienza esatta”. Ovvero, non sono disponibili procedure valide in tutti i casi e non sempre siamo in grado di rispondere alle domande precedenti. È piuttosto l'esperienza del ricercatore che consente di riconoscere una “buona” catena di Markov e a suggerire cosa si può fare per riparare una “cattiva” catena di Markov. In questo Capitolo ci concentreremo su alcuni strumenti diagnostici grafici e numerici che possono essere utilizzati per la diagnostica delle catene markoviane. L'utilizzo di questi strumenti diagnostici deve essere eseguito in modo olistico. Nessuna singola diagnostica visiva o numerica è infatti sufficiente: un quadro completo della qualità della catena di Markov si può solo ottenere considerando tutti gli strumenti descritti di seguito.

2.1 Esame dei *trace plot*

La convergenza e il “mixing” possono essere controllate mediante il *trace plot* che mostra l'andamento delle simulazioni e ci dice se stiamo effettivamente

utilizzando una distribuzione limite. Consideriamo nuovamente il *trace plot* del simulazione Beta-Binomiale della figura 2.1:

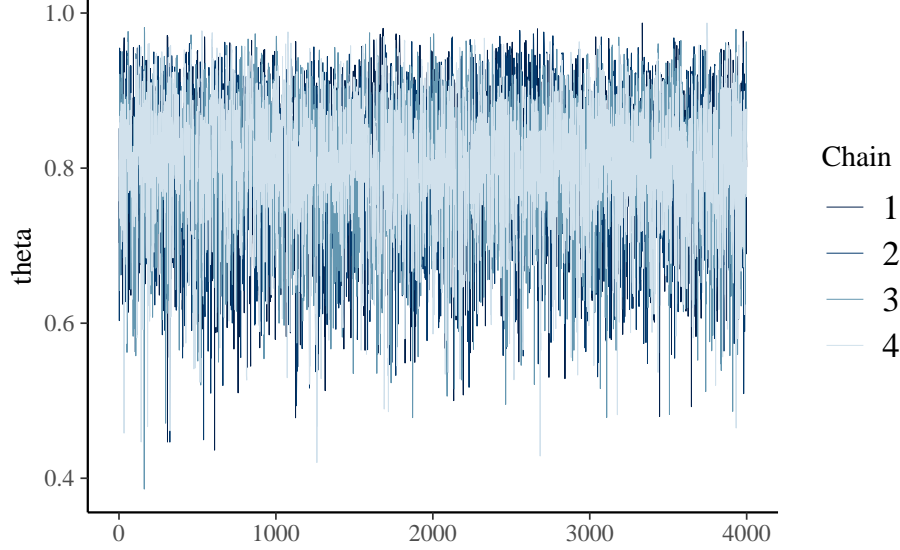


Figura 2.1: Trace plot per il modello Beta-Binomiale dei dati di Gautret et al. (2020).

La figura 2.1 fornisce un esempio perfetto di come dovrebbero apparire i *trace plot*. Quando le catene markoviane raggiungono uno stato stazionario e sono stabili ciò significa che hanno raggiunto la distribuzione stazionaria e il *trace plot* rivela una assenza di struttura e assomiglia alla rappresentazione del rumore bianco, come nella figura 2.1. Al contrario, la figura 2.2 indica mancanza di convergenza¹.

Consideriamo i trace-plot della figura 2.2 (a sinistra). La tendenza verso il basso indica che la catena A non è stazionaria, ovvero non si mantiene costante all’evolversi nel tempo. La tendenza verso il basso suggerisce inoltre la presenza di una forte correlazione tra i valori della catena: il trace-plot non fornisce una rappresentazione di rumore indipendente. Tutto questo significa che la catena A “si sta mescolando lentamente”. Sebbene le catene di Markov siano intrinsecamente dipendenti, più si comportano come se fossero dei campioni casuali (rumorosi), minore è l’errore dell’approssimazione alla distribuzione a posteriori. La catena B presenta un problema diverso. Come evidenziato dalle due linee completamente piatte nel tracciato, essa tende a bloccarsi quando visita valori bassi di θ .

Gli istogrammi lisciati della figura 2.2 (a destra) confermano che entrambe queste catene sono problematiche: infatti producono approssimazioni scadenti della distribuzione a posteriori che, nell’esempio di Johnson et al. (2022), è una Beta(11,3) (curva nera nella figura). Consideriamo la catena A. Dal momento che si sta mescolando lentamente, nelle iterazioni eseguite ha esplorato unicamente i valori θ nell’intervallo da 0.6 a 0.9. Di conseguenza, la sua approssimazione della distribuzione a posteriori sopravvaluta la plausibilità dei valori θ in questo intervallo e, nel contempo, sottovaluta la plausibilità dei valori θ esterni a questo intervallo. Consideriamo ora la catena B. Rimanendo bloccata, la catena

¹Figura riprodotta da Johnson et al. (2022)

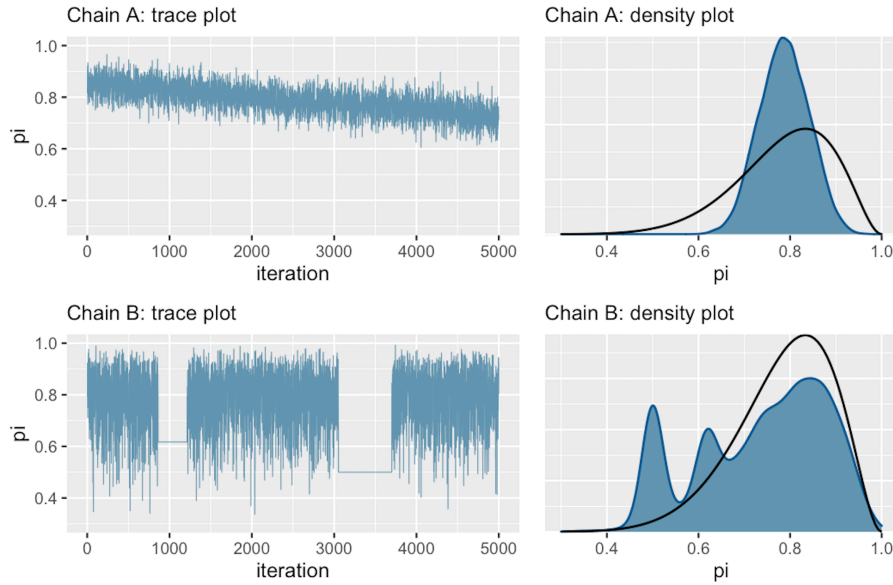


Figura 2.2: Trace plots (a sinistra) e corrispondenti grafici di densità (a destra) di due ipotetiche catene di Markov. Queste figure forniscono due esempi di come potrebbero apparire delle catene di Markov non stazionarie. Le linee nere sovrapposte alle densità empiriche (a destra) rappresentano una ipotetica distribuzione target $Beta(11,3)$.

B campiona in maniera eccessiva alcuni valori nella coda sinistra della distribuzione a posteriori di θ . Questo fenomeno produce i picchi che sono presenti nell'approssimazione alla distribuzione a posteriori.

In pratica, al di là dei presenti esempi “scolastici” (in cui disponiamo di una formulazione analitica della distribuzione a posteriori), non abbiamo mai il privilegio di poter confrontare i risultati del campionamento MCMC con la corretta distribuzione a posteriori. Ecco perché la diagnostica delle catene di Markov è così importante: se vediamo trace-plots come quelli della figura 2.2, sappiamo che non abbiamo ottenuto una adeguata approssimazione della distribuzione a posteriori.

In tali circostanze possiamo ricorrere ad alcuni rimedi.

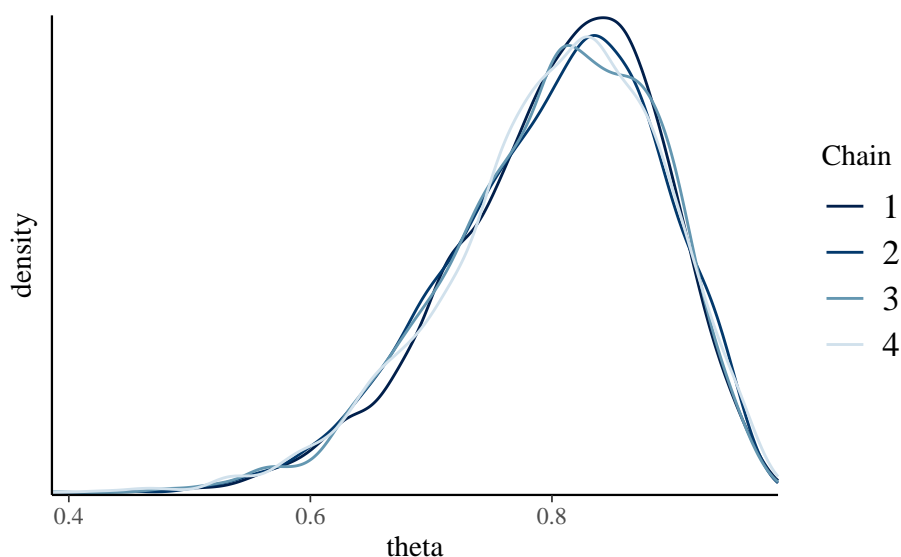
1. Controllare il modello. Siamo sicuri che le distribuzioni a priori e la verosimiglianza siano appropriate per i dati osservati?
2. Utilizzare un numero maggiore di iterazioni. Alcune tendenze indesiderate a breve termine della catena possono appianarsi nel lungo termine.

2.2 Confronto delle catene parallele

Nella simulazione `cmdstanr()` per il modello Beta-Binomiale dei dati di Gautret et al. (2020) abbiamo utilizzato quattro catene di Markov parallele. Non solo è necessario che ogni singola catena sia stazionaria (come discusso sopra), ma è anche necessario che le quattro catene siano coerenti tra loro. Sebbene le catene esplorino percorsi diversi nello spazio dei parametri, quando convergono ad uno stato di equilibrio dovrebbero presentare caratteristiche simili e dovrebbero

produrre approssimazioni simili alla distribuzione a posteriori. Per l'esempio del modello Beta-Binomiale dei dati di Gautret et al. (2020), i seguenti istogrammi lisciati indicano che le quattro catene producono approssimazioni quasi indistinguibili della distribuzione a posteriori. Ciò prova che la simulazione è stabile e contiene un numero sufficiente di valori: l'esecuzione delle catene per un numero maggiore di iterazioni non migliorerebbe l'approssimazione della distribuzione a posteriori.

```
mcmc_dens_overlay(stanfit1, pars = "theta") +  
  ylab("density")
```



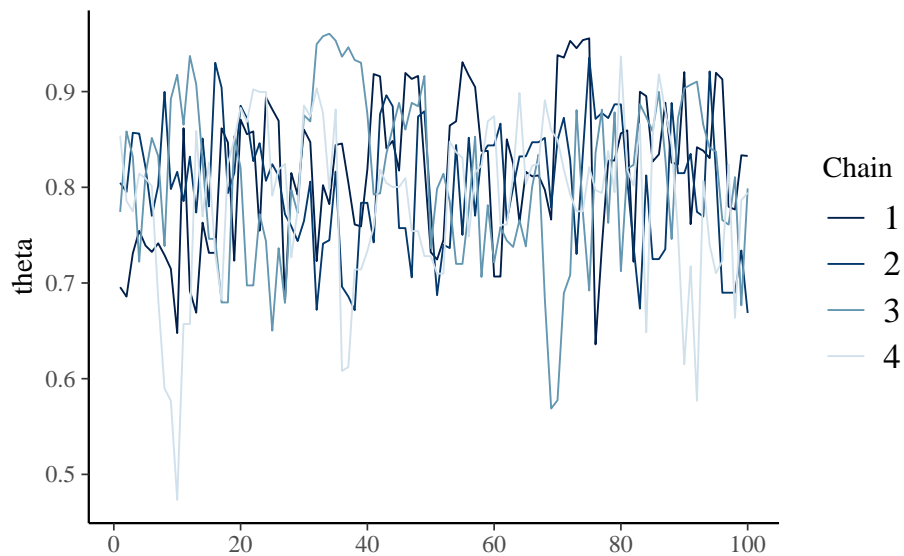
Per fare un confronto, consideriamo la simulazione di una catena di Markov più corta per lo stesso modello. La chiamata seguente richiede quattro catene parallele per sole 100 iterazioni ciascuna:

```
bb_short <- mod$sample(  
  data = data1_list,  
  iter_sampling = 50 * 2L,  
  seed = SEED,  
  chains = 4L,  
  parallel_chains = 4L,  
  refresh = 0,  
  thin = 1  
)  
FALSE Running MCMC with 4 parallel chains...  
FALSE  
FALSE Chain 1 finished in 0.0 seconds.  
FALSE Chain 2 finished in 0.0 seconds.  
FALSE Chain 3 finished in 0.0 seconds.  
FALSE Chain 4 finished in 0.0 seconds.  
FALSE  
FALSE All 4 chains finished successfully.
```

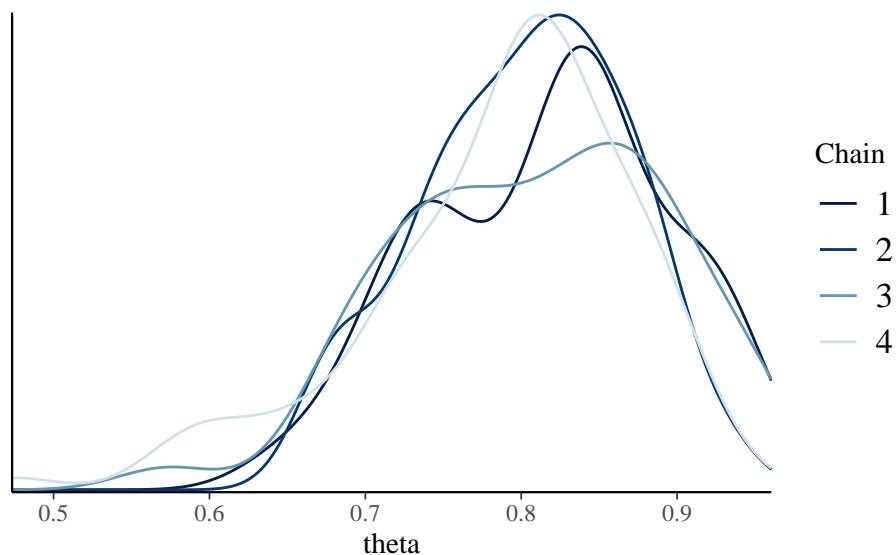
```
FALSE Mean chain execution time: 0.0 seconds.  
FALSE Total execution time: 0.2 seconds.  
  
stanfit_bb_short <- rstan::read_stan_csv(bb_short$output_files())
```

Di seguito sono riportati i *trace-plot* e i corrispondenti istogrammi lisciati.

```
mcmc_trace(stanfit_bb_short, pars = "theta")
```



```
mcmc_dens_overlay(stanfit_bb_short, pars = "theta")
```



Anche se i *trace plot* sembrano tutti mostrare un andamento casuale, gli istogrammi lisciati sono piuttosto diversi tra loro e producono approssimazioni diverse della distribuzione a posteriori. Di fronte a tale instabilità è chiaro che sarebbe un errore interrompere la simulazione dopo solo 100 iterazioni.

2.3 Numerosità campionaria effettiva

Nella simulazione del modello Beta-Binomiale per i dati di Gautret et al. (2020) abbiamo utilizzato quattro catene di Markov parallele che producono un totale di $N = 16000$ campioni *dipendenti* di θ . Sapendo che l'errore dell'approssimazione alla distribuzione a posteriori è probabilmente più grande di quello che si otterrebbe usando 16000 campioni *indipendenti*, ci possiamo porre la seguente domanda: quanti campioni indipendenti sarebbero necessari per produrre un'approssimazione della distribuzione a posteriori equivalentemente a quella che abbiamo ottenuto? La numerosità campionaria effettiva (*effective sample size*, N_{eff}) fornisce una risposta a questa domanda.

Tipicamente, $N_{eff} < N$, per cui il rapporto campionario effettivo (*effective sample size ratio*) $\frac{N_{eff}}{N}$ è minore di 1. Come regola euristica, viene considerato problematico un rapporto campionario effettivo minore del 10% del numero totale di campioni ottenuti nella simulazione (più basso è il rapporto campionario effettivo peggiore è il “mixing” della catena). La funzione `bayesplot::neff_ratio()` consente di calcolare il rapporto campionario effettivo. Per il modello Beta-Binomiale dei dati di Gautret et al. (*ibid.*), questo rapporto è di circa 0.34:

```
bayesplot::neff_ratio(stanfit1, pars = c("theta"))
#> [1] 0.3352141
```

Ciò indica che l'accuratezza dell'approssimazione della distribuzione a posteriori di θ ottenuta mediante 16000 campioni dipendenti è approssimativamente simile a quella che si potrebbe ottenere con

```
bayesplot::neff_ratio(stanfit1, pars = c("theta")) * 16000
#> [1] 5363.426
```

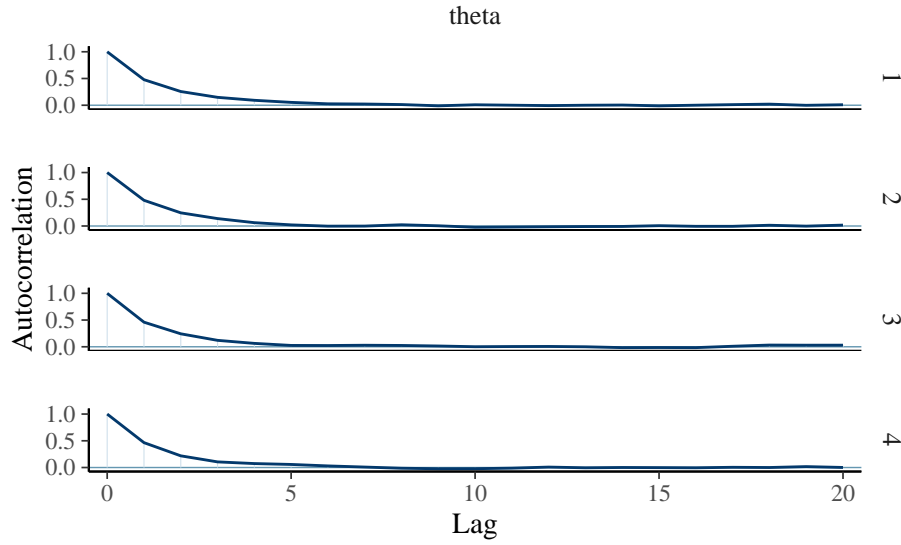
campioni *indipendenti*. In questo esempio, il rapporto campionario effettivo è maggiore di 0.1; dunque non ci sono problemi.

2.4 Autocorrelazione

Normalmente un algoritmo MCMC genera catene di Markov di campioni, ognuno dei quali è autocorrelato a quelli generati immediatamente prima e dopo di lui. Conseguentemente campioni successivi non sono indipendenti ma formano una catena di Markov con un certo grado di correlazione. Il valore $\theta^{(i)}$ tende ad essere più simile al valore $\theta^{(i-1)}$ che al valore $\theta^{(i-2)}$, o al valore $\theta^{(i-3)}$, eccetera. Una misura di ciò è fornita dall'autocorrelazione tra i valori consecutivi della catena.

Il correlogramma per ciascuna delle quattro catene dell'esempio si produce con la seguente chiamata:

```
bayesplot::mcmc_acf(stanfit1, pars = "theta")
```



Il correlogramma mostra l'autocorrelazione in funzione di ritardi da 0 a 20. L'autocorrelazione di lag 0 è naturalmente 1 – misura la correlazione tra un valore della catena di Markov e se stesso. L'autocorrelazione di lag 1 è di circa 0.5, indicando una correlazione moderata tra i valori della catena che distano di solo 1 passo l'uno dall'altro. Successivamente, l'autocorrelazione diminuisce rapidamente ed è effettivamente pari a 0 per un lag di 5. Questo risultato fornisce una conferma del fatto che la catena di Markov costituisce una buona approssimazione di un campione casuale di $p(\theta | y)$.

Al contrario, nella figura 2.3 (a destra) (riprodotta da Johnson et al. 2022) vediamo un esempio nel quale il trace plot rivela una forte tendenza tra i valori di una catena di Markov e, dunque, una forte autocorrelazione.

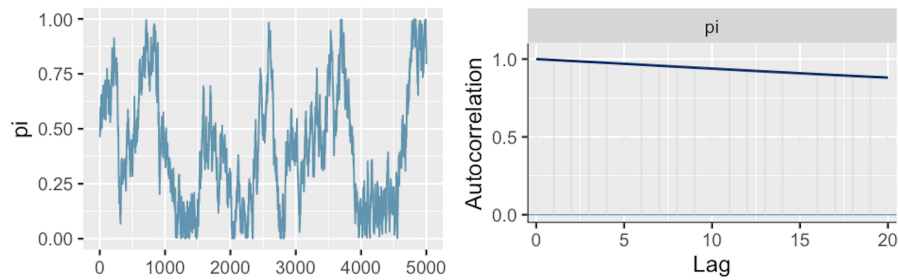


Figura 2.3: Trace plot (a sinistra) e correlogramma (a destra) di una catena di Markov in cui il *mixing* è lento – figura riprodotta da @Johnson2022bayesrules.

Questa osservazione è confermata nell'correlogramma (a destra). La lenta diminuzione della curva di autocorrelazione indica che la dipendenza tra i valori della catena non svanisce rapidamente. Con un lag di 20 la correlazione è addirittura pari a 0.9. Poiché i valori della catena sono fortemente associati tra loro, il “mixing” è lento: la simulazione richiede un numero molto grande di iterazioni per esplorare adeguatamente l'intera gamma di valori della distribuzione a posteriori.²

In presenza di catene di Markov non *rapidly mixing* sono possibili due rimedi.

²Una (famiglia di) catene di Markov è *rapidly mixing* se mostra un comportamento simile

- Aumentare il numero di iterazioni. Anche una catena non *rapidly mixing* può produrre eventualmente una buona approssimazione della distribuzione a posteriori se il numero di iterazioni è sufficientemente grande.
- *Thinning*. Per esempio, se la catena di Markov è costituita da 16000 valori di θ , potremmo decidere di conservare solo ogni secondo valore e ignorare gli altri valori: $\{\theta^{(2)}, \theta^{(4)}, \theta^{(6)}, \dots, \theta^{(16000)}\}$. Oppure, potremmo decidere di conservare ogni decimo valore: $\{\theta^{(10)}, \theta^{(20)}, \theta^{(30)}, \dots, \theta^{(16000)}\}$. Scartando i campioni intermedi, è possibile rimuovere le forti correlazioni che sono presenti nel caso di lag più piccoli.

Vediamo ora come sia possibile estrarre i valori di una catena dall'oggetto `stanfit1`.

```
# valori delle 4 catene
S <- ggmcmc::ggs(stanfit1)
head(S)
#> # A tibble: 6 x 4
#>   Iteration Chain Parameter value
#>   <dbl> <int> <fct>    <dbl>
#> 1     1     1 1 theta    0.833
#> 2     2     1 1 theta    0.822
#> 3     3     1 1 theta    0.633
#> 4     4     1 1 theta    0.798
#> 5     5     1 1 theta    0.855
#> 6     6     1 1 theta    0.909
```

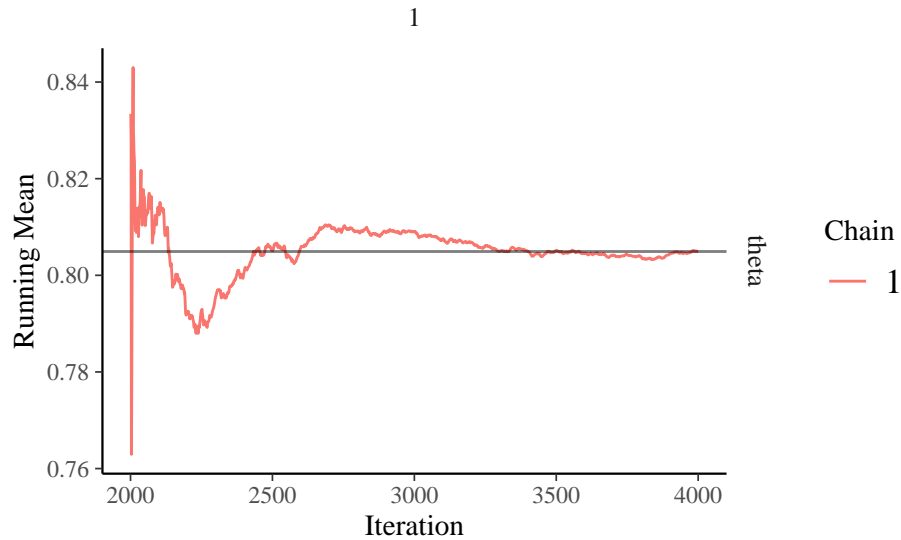
La prima catena può essere isolata nel modo seguente:

```
S1 <- S %>%
  dplyr::filter(
    Chain == 1,
    Parameter == "theta"
  )
```

Una serie temporale della catena si ottiene con la funzione `ggmcmc::ggs_running`:

```
ggmcmc::ggs_running(S1)
```

a quello di un campione indipendente: i valori delle catene si addensano nell'intervallo dei valori più plausibili della distribuzione a posteriori; l'autocorrelazione tra i valori della catena diminuisce rapidamente; il rapporto campionario effettivo è ragionevolmente grande. Le catene che non sono *rapidly mixing* non godono delle caratteristiche di un campione indipendente: le catene non si addensano nell'intervallo dei valori più plausibili della distribuzione a posteriori; l'autocorrelazione tra i valori della catena diminuisce molto lentamente; il rapporto campionario effettivo è piccolo.



Il grafico precedente mostra che, per il modello bayesiano che stiamo discutendo, una condizione di equilibrio della catena di Markov richiederebbe un numero maggiore di iterazioni di quelle che sono state effettivamente simulate.

L'autocorrelazione di ordine 1 si ottiene nel modo seguente (si veda il Paragrafo ??):

```
cor(S1$value[-length(S1$value)], S1$value[-1])
#> [1] 0.471083
```

Questo valore corrisponde a ciò che è riportato nel correlogramma mostrato sopra.

2.5 Statistica \hat{R}

In precedenza abbiamo detto che non solo è necessario che ogni singola catena sia stazionaria, è anche necessario che le diverse catene siano coerenti tra loro. La statistica \hat{R} affronta questo problema calcolando il rapporto tra la varianza tra le catene markoviane e la varianza entro le catene. In una situazione ottimale $\hat{R} = 1$; se \hat{R} è lontano da 1 questo vuol dire che non è ancora stata raggiunta la convergenza.

È possibile calcolare \hat{R} mediante la chiamata alla funzione `bayesplot::rhat()`. Per il modello Beta-Binomiale applicato ai dati di Gautret et al. (2020) abbiamo:

```
bayesplot::rhat(stanfit1, pars = "theta")
#> [1] 1.000499
```

il che indica che il valore \hat{R} ottenuto è molto simile al valore ottimale.

In maniera euristica, si può affermare che se \hat{R} supera la soglia di 1.05 questo viene interpretato come evidenza che le diverse catene parallele non producono approssimazioni coerenti della distribuzione a posteriori, quindi la simulazione è instabile.

```
ggmcmc::ggs_Rhat(S) + xlab("R_hat") + xlim(0.95, 1.05)
```



Utilizzando l'oggetto `stanfit1`, possiamo recuperare la statistica di Geweke nel modo seguente:


```
#>
#> Fraction in 1st window = 0.1
#> Fraction in 2nd window = 0.5
#>
#> theta
#> 0.0759
#>
#>
#> [[3]]
#>
#> Fraction in 1st window = 0.1
#> Fraction in 2nd window = 0.5
#>
#> theta
#> -0.7616
#>
#>
#> [[4]]
#>
#> Fraction in 1st window = 0.1
#> Fraction in 2nd window = 0.5
#>
#> theta
#> 1.878
```

Per interpretare questi valori ricordiamo che la statistica di Geweke è uguale a zero quando le medie delle due porzioni della catena di Markov sono uguali. Valori maggiori di $|2|$ suggeriscono che la catena non ha ancora raggiunto una distribuzione stazionaria.

Sintesi a posteriori

La distribuzione a posteriori è un modo per descrivere il nostro grado di incertezza rispetto al parametro incognito (o rispetto ai parametri incogniti) oggetto dell'inferenza. La distribuzione a posteriori contiene tutte le informazioni disponibili sui possibili valori del parametro. Se il parametro esaminato è monodimensionale (o bidimensionale) è possibile fornire un grafico di tutta la distribuzione a posteriori $p(\theta | y)$. Tuttavia, spesso vogliamo anche giungere ad una sintesi numerica della distribuzione a posteriori, soprattutto se il vettore dei parametri ha più di due dimensioni. A a questo proposito è possibile utilizzare le consuete statistiche descrittive, come media, mediana, moda, varianza, deviazione standard e i quantili. In alcuni casi, queste statistiche descrittive sono più facili da presentare e interpretare rispetto alla rappresentazione grafica della distribuzione a posteriori.

La stima puntuale della tendenza centrale della distribuzione a posteriori fornisce informazioni su quello che può essere considerato come il “valore più plausibile” del parametro. L'intervallo di credibilità fornisce invece un'indicazione dell'ampiezza dell'intervallo che contiene una determinata quota della massa della distribuzione a posteriori del parametro.

3.1 Stima puntuale

Per sintetizzare la distribuzione a posteriori in modo da giungere ad una stima puntuale di θ si è soliti scegliere tra moda, mediana o media a seconda del tipo di distribuzione con cui si ha a che fare e della sua forma. Ogni stima puntuale ha una sua interpretazione.

- La media è il valore atteso a posteriori del parametro.
- La moda può essere interpretata come il singolo valore più credibile (“più plausibile”) del parametro, alla luce dei dati, ovvero il valore per il parametro θ che massimizza la distribuzione a posteriori. Per questa ragione la moda viene detta *massimo a posteriori*, MAP. Il limite della moda quale

statistica riassuntiva della distribuzione a posteriori è che, talvolta, tale distribuzione è multimodale e il MAP non è necessariamente il valore “più credibile”.

- La mediana è il valore del parametro tale per cui, su entrambi i lati di essa, giace il 50% della massa di probabilità a posteriori.

La misura di variabilità del parametro è la *varianza a posteriori* la quale, nel caso di una distribuzione a posteriori ottenuta per via numerica, si calcola con la formula della varianza che conosciamo rispetto alla tendenza centrale data dalla media a posteriori. La radice quadrata della varianza a posteriori è la *deviazione standard a posteriori* che descrive l'incertezza a posteriori circa il parametro di interesse nella stessa unità di misura dei dati.

Le procedure bayesiane basate sui metodi MCMC utilizzano un numero finito di campioni dalla distribuzione stazionaria, e una tale caratteristica della simulazione introduce un ulteriore livello di incertezza nella stima del parametro. L'*errore standard della stima* (in inglese *Monte Carlo standard error*, MCSE) misura l'accuratezza della simulazione. La deviazione standard a posteriori e l'errore standard della stima sono due concetti completamente diversi. La deviazione standard a posteriori descrive l'incertezza circa il parametro (l'ampiezza della distribuzione a posteriori) ed è una funzione della dimensione del campione; il MCSE descrive invece l'incertezza nella stima del parametro dovuta alla simulazione MCMC ed è una funzione del numero di iterazioni nella simulazione.

3.2 Intervallo di credibilità

Molto spesso la stima puntuale è accompagnata da una stima intervallare. Nella statistica bayesiana, se il parametro $\theta \in \Theta$ è monodimensionale, si dice *intervallo di credibilità* un intervallo di valori I_α che contiene la proporzione $1 - \alpha$ della massa di probabilità della funzione a posteriori:

$$p(\theta \in I_\alpha \mid y) = 1 - \alpha. \quad (3.1)$$

L'intervallo di credibilità ha lo scopo di esprimere il nostro grado di incertezza riguardo la stima del parametro. Se il parametro θ è multidimensionale, si parla invece di “regione di credibilità”.

La condizione (3.1) non determina un unico intervallo di credibilità al $(1 - \alpha) \cdot 100\%$. In realtà esiste un numero infinito di tali intervalli. Ciò significa che dobbiamo definire alcune condizioni aggiuntive per la scelta dell'intervallo di credibilità. Esaminiamo due delle condizioni aggiuntive più comuni.

Intervallo di credibilità a code uguali

Un intervallo di credibilità *a code uguali* a livello α è un intervallo

$$I_\alpha = [q_{\alpha/2}, 1 - q_{\alpha/2}],$$

dove q_z è un quantile z della distribuzione a posteriori. Per esempio, l'intervallo di credibilità a code uguali al 95% è un intervallo

$$I_{0.05} = [q_{0.025}, q_{0.975}]$$

che lascia il 2.5% della massa di densità a posteriori in ciascuna coda.

Intervallo di credibilità a densità a posteriori più alta

Nell'intervallo di credibilità a code uguali alcuni valori del parametro che sono inclusi nell'intervallo possono avere una credibilità a posteriori più bassa rispetto a quelli esterni all'intervallo. L'intervallo di credibilità *a densità a posteriori più alta* (in inglese *High Posterior Density Interval*, HPD) è invece costruito in modo tale da assicurare di includere nell'intervallo tutti i valori θ che sono a posteriori maggiormente credibili. Graficamente questo intervallo può essere ricavato tracciando una linea orizzontale sulla rappresentazione della distribuzione a posteriori e regolando l'altezza della linea in modo tale che l'area sottesa alla curva sia pari a $1 - \alpha$. Questo tipo di intervallo è il più stretto possibile, tra tutti i possibili intervalli di credibilità allo stesso livello di fiducia. Se la distribuzione a posteriori è simmetrica unimodale, l'intervallo di credibilità a densità a posteriori più alta corrisponde all'intervallo di credibilità a code uguali.

Interpretazione

L'interpretazione dell'intervallo di credibilità è molto intuitiva: l'intervallo di credibilità è un intervallo di valori all'interno del quale cade il valore del parametro incognito con un particolare livello di probabilità soggettiva. Possiamo dire che, dopo aver visto i dati crediamo, con un determinato livello di probabilità soggettiva, che il valore del parametro (ad esempio, la dimensione dell'effetto di un trattamento) abbia un valore compreso all'interno dell'intervallo che è stato calcolato, laddove per probabilità soggettiva intendiamo “il grado di fiducia che lo sperimentatore ripone nel verificarsi di un evento”. Gli intervalli di credibilità si calcolano con un software.

3.3 Un esempio concreto

Per fare un esempio pratico, consideriamo nuovamente i valori del BDI-II dei 30 soggetti clinici di Zetsche et al. (2019):

```
suppressPackageStartupMessages(library("bayesrules"))

df <- data.frame(
  y = c(
    26, 35, 30, 25, 44, 30, 33, 43, 22, 43,
    24, 19, 39, 31, 25, 28, 35, 30, 26, 31,
    41, 36, 26, 35, 33, 28, 27, 34, 27, 22
  )
)
```

Un valore BDI-II ≥ 30 indica la presenza di un livello “grave” di depressione. Nel campione clinico di Zetsche et al. (*ibid.*),

```
sum(df$y > 29)
#> [1] 17
```

17 pazienti su 30 manifestano un livello grave di depressione.

Supponiamo di volere stimare la distribuzione a posteriori della probabilità θ di depressione “grave” nei pazienti clinici, così come viene misurata dal test BDI-II, imponendo su θ una distribuzione a priori $\text{Beta}(8, 2)$.

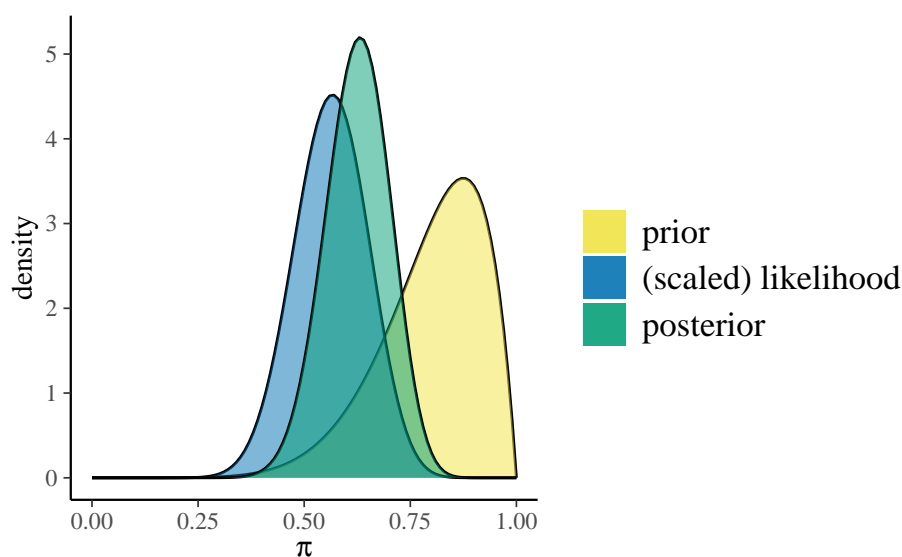
Sappiamo che il modello Beta-Binomiale può essere espresso nella forma seguente:

$$\begin{aligned} Y|\theta &\sim \text{Bin}(30, \theta) \\ \theta &\sim \text{Beta}(8, 2) \end{aligned}$$

con una corrispondente distribuzione a posteriori $\text{Beta}(25, 15)$:

$$f(\theta|y = 17) = \frac{\Gamma(25 + 15)}{\Gamma(25)\Gamma(15)} \theta^{25-1} (1 - \theta)^{15-1} \quad \text{for } \theta \in [0, 1]. \quad (3.2)$$

```
plot_beta_binomial(alpha = 8, beta = 2, y = 17, n = 30)
```



Stime puntuali della distribuzione a posteriori

Una volta trovata l'intera distribuzione a posteriori, quale valore di sintesi è necessario riportare? Questa sembra una domanda innocente, ma in realtà è una domanda a cui è difficile rispondere. La stima bayesiana dei parametri è fornita dall'intera distribuzione a posteriori, che non è un singolo numero, ma una funzione che mappa ciascun valore del parametro ad un valore di plausibilità. Quindi non è necessario scegliere una stima puntuale. In linea di principio, una stima puntuale non è quasi mai necessaria ed è spesso dannosa in quanto comporta una perdita di informazioni.

Tuttavia talvolta una tale sintesi è richiesta. Diverse risposte sono allora possibili. La media della distribuzione a posteriori per θ è

$$\mathbb{E}(\pi \mid y = 17) = \frac{\alpha}{\alpha + \beta} = \frac{25}{25 + 15} = 0.625.$$

Una stima del massimo della probabilità a posteriori, o brevemente massimo a posteriori, MAP (da *maximum a posteriori probability*), è la moda della distribuzione a posteriori. Nel caso presente, una stima del MAP può essere ottenuta nel modo seguente:

$$\text{Mo}(\pi \mid y = 17) = \frac{\alpha - 1}{\alpha + \beta - 2} = \frac{25 - 1}{25 + 15 - 2} = 0.6316.$$

Gli stessi risultati si ottengono usando la chiamata a `bayesrules::summarize_beta_binomial()`:

```
summarize_beta_binomial(alpha = 8, beta = 2, y = 17, n = 30)
#>      model alpha beta mean    mode      var
#> 1   prior      8    2 0.800 0.8750000 0.014545455
#> 2 posterior    25   15 0.625 0.6315789 0.005716463
#>      sd
#> 1 0.1206045
#> 2 0.0756073
```

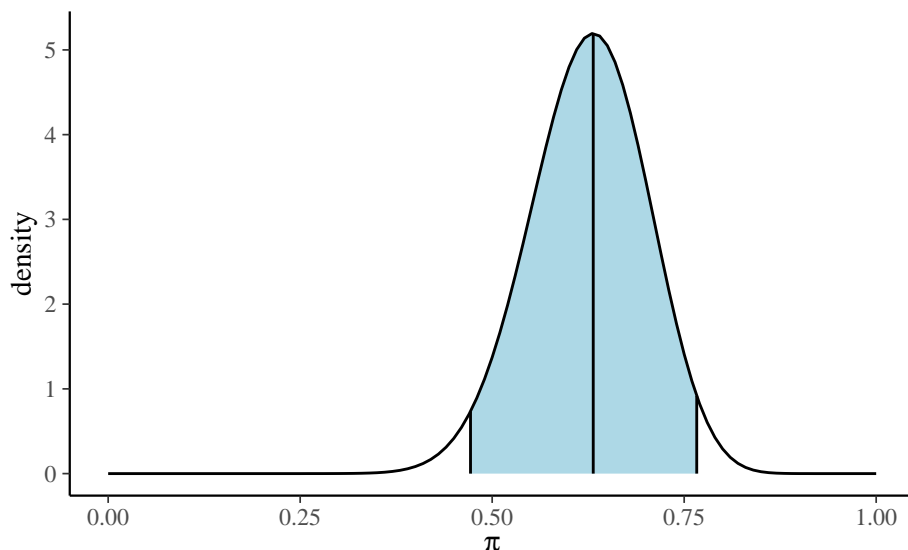
La mediana si ottiene con

```
qbeta(.5, shape1 = 25, shape2 = 15)
#> [1] 0.6271031
```

Intervallo di credibilità

È più comune sintetizzare la distribuzione a posteriori mediante l'intervallo di credibilità. Per esempio, l'intervallo di credibilità a code uguali al 95%

```
plot_beta_ci(alpha = 25, beta = 15, ci_level = 0.95)
```



è dato dalla chiamata

```
qbeta(c(0.025, 0.975), 25, 15)
#> [1] 0.4717951 0.7663607
```

Il calcolo precedente evidenzia l'interpretazione intuitiva dell'intervallo di credibilità. Tale intervallo, infatti, può essere interpretato come la probabilità che θ assuma valori compresi tra 0.472 e 0.766:

$$P(\theta \in (0.472, 0.766) | Y = 17) = \int_{0.472}^{0.766} f(\theta | y = 17) d\theta = 0.95,$$

ovvero

```
postFun <- function(theta) {
  gamma(25 + 15) / (gamma(25) * gamma(15)) * theta^24 * (1 - theta)^14
}
integrate(
  postFun,
  lower = 0.4717951,
  upper = 0.7663607
)$value
#> [1] 0.95
```

Possiamo costruire diversi intervalli di credibilità a code equivalenti. Ad esempio, l'intervallo di credibilità compreso tra il 25-esimo e il 75-esimo percentile è

```
qbeta(c(0.25, 0.75), 25, 15)
#> [1] 0.5743878 0.6778673
```

ovvero, abbiamo una certezza a posteriori del 50% che la probabilità di depressione grave tra i pazienti clinici sia un valore compreso tra 0.57 e 0.68.

Non esiste un livello credibile “corretto”. I ricercatori, utilizzano vari livelli, ad esempio 50%, 80% o 95%, a seconda del contesto dell'analisi. Ciascuno di questi intervalli fornisce un'immagine diversa della nostra comprensione della distribuzione a posteriori del parametro di interesse.

Non è inoltre necessario riportare l'intervallo di credibilità a code uguali. Se la distribuzione a posteriori è fortemente asimmetrica è più sensato riportare l'intervallo di credibilità a densità a posteriori più alta. L'intervallo HPD risulta più semplice da determinare quando la distribuzione a posteriori viene approssimata con il metodo MCMC.

Probabilità della distribuzione a posteriori

Il test di ipotesi è un compito comune dell'analisi della distribuzione a posteriori. Supponiamo che si voglia conoscere la probabilità a posteriori che θ sia superiore a 0.5. Per sapere quanto credibile sia l'evento $\theta > 0.5$ possiamo calcolare il seguente integrale:

$$P(\theta > 0.5 \mid y = 17) = \int_{0.5}^1 f(\theta \mid y = 17) d\theta,$$

dove $f(\cdot)$ è la distribuzione $(25, 15)$:

```
pbeta(0.5, shape1 = 25, shape2 = 15, lower.tail = FALSE)
#> [1] 0.9459355
```

il che è equivalente a:

```
postFun <- function(theta) {
  gamma(25 + 15) / (gamma(25) * gamma(15)) * theta^24 * (1 - theta)^14
}
integrate(
  postFun,
  lower = 0.5,
  upper = 1
)$value
#> [1] 0.9459355
```

È anche possibile formulare un test di ipotesi contrastando due ipotesi contrapposte. Per esempio, $H_1 : \theta \geq 0.5$ e $H_2 : \theta < 0.5$. Ciò consente di calcolare l'*odds a posteriori* di $\theta > 0.5$:

$$\text{poterior odds} = \frac{H_1 \mid y = 17}{H_2 \mid y = 17} \quad (3.3)$$

ovvero

```
posterior_odds <-
  pbeta(0.5, shape1 = 25, shape2 = 15, lower.tail = FALSE) /
  pbeta(0.5, shape1 = 25, shape2 = 15, lower.tail = TRUE)
posterior_odds
#> [1] 17.49642
```

L'*odds a posteriori* rappresenta l'aggiornamento delle nostre credenze dopo avere osservato $y = 17$ in $n = 30$. L'*odds a priori* di $\theta > 0.5$ era:

```
prior_odds <-
  pbeta(0.5, shape1 = 8, shape2 = 2, lower.tail = FALSE) /
  pbeta(0.5, shape1 = 8, shape2 = 2, lower.tail = TRUE)
prior_odds
#> [1] 50.2
```

Il *fattore di Bayes* (*Bayes Factor*; BF) confronta gli *odds a posteriori* con gli *odds a priori* e quindi fornisce informazioni su quanto sia mutata la nostra comprensione relativa a θ dopo avere osservato i nostri dati del campione:

$$BF = \frac{\text{odds a posteriori}}{\text{odds a priori}}.$$

Nel caso presente abbiamo

```
BF <- posterior_odds / prior_odds
BF
#> [1] 0.3485343
```

Quindi, dopo avere osservato i dati, gli odds a posteriori della nostra ipotesi a proposito di θ sono pari a solo il 34% degli odds a priori.

Per fare un altro esempio, consideriamo invece il caso in cui le credenze a priori rivelano una credenza diametralmente opposta rispetto a θ che nel caso considerato in precedenza, ovvero $\text{Beta}(2, 8)$. In questo secondo caso, la distribuzione a posteriori diventa

```
summarize_beta_binomial(alpha = 2, beta = 8, y = 17, n = 30)
#>      model alpha beta mean      mode      var
#> 1   prior      2   8 0.200 0.1250000 0.014545455
#> 2 posterior    19  21 0.475 0.4736842 0.006082317
#>      sd
#> 1 0.12060454
#> 2 0.07798921
```

e il BF è

```
posterior_odds <-
  pbeta(0.5, shape1 = 19, shape2 = 21, lower.tail = FALSE) /
  pbeta(0.5, shape1 = 19, shape2 = 21, lower.tail = TRUE)

prior_odds <-
  pbeta(0.5, shape1 = 2, shape2 = 8, lower.tail = FALSE) /
  pbeta(0.5, shape1 = 2, shape2 = 8, lower.tail = TRUE)

BF <- posterior_odds / prior_odds
BF
#> [1] 30.07239
```

In altre parole, in questo secondo esempio gli odds a posteriori della nostra ipotesi a proposito di θ sono aumentati di 30 volte rispetto agli odds a priori.

In generale, in un test di ipotesi che contrappone un'ipotesi sostantiva H_a ad un'ipotesi nulla H_0 il BF è un rapporto di odds per l'ipotesi sostantiva:

$$\text{Bayes Factor} = \frac{\text{posterior odds}}{\text{prior odds}} = \frac{P(H_a | Y)/P(H_0 | Y)}{P(H_a)/P(H_0)}.$$

Essendo un rapporto, il BF deve essere valutato rispetto al valore di 1. Ci sono tre possibilità:

- $BF = 1$: La credibilità di H_a non è cambiata dopo avere osservato i dati.
- $BF > 1$: La credibilità di H_a è aumentata dopo avere osservato i dati.
Quindi maggiore è BF , più convincente risulta l'evidenza per H_a .
- $BF < 1$: La credibilità di H_a è diminuita dopo avere osservato i dati.

Non ci sono delle soglie universalmente riconosciute per interpretare il BF . Per esempio, Lee e Wagenmakers (2014) propongono il seguente schema:

BF	Interpretation
> 100	Extreme evidence for H_a
$30 - 100$	Very strong evidence for H_a
$10 - 30$	Strong evidence for H_a
$3 - 10$	Moderate evidence for H_a
$1 - 3$	Anecdotal evidence for H_a
1	No evidence
$1/3 - 1$	Anecdotal evidence for H_0
$1/10 - 1/3$	Moderate evidence for H_0
$1/30 - 1/10$	Strong evidence for H_0
$1/100 - 1/30$	Very strong evidence for H_0
$< 1/100$	Extreme evidence for H_0

Tuttavia, è importante notare che l'opinione maggiormente diffusa nella comunità scientifica sia quella che incoraggia a non trarre conclusioni rigide dai dati utilizzando dei criteri fissati una volta per tutte. Pertanto, non esiste una soglia univoca per il BF che consente di classificare le ipotesi dei ricercatori nelle due categorie “vero” o “falso”. Invece, è più utile adottare una pratica più flessibile capace di tenere in considerazione il contesto e le potenziali implicazioni di ogni singolo test di ipotesi. Inoltre, è stato molte volte ripetuto che la distribuzione a posteriori è molto più informativa di una decisione binaria: la rappresentazione di tutta la distribuzione a posteriori fornisce una misura olistica del nostro livello di incertezza riguardo all'affermazione (il parametro, ovvero l'ipotesi) che viene valutata.

Considerazioni conclusive

Questo capitolo introduce le procedure di base per la manipolazione della distribuzione a posteriori. Lo strumento fondamentale che è stato utilizzato è quello fornito dai campioni di valori del parametro che vengono estratti dalla distribuzione a posteriori. Lavorare con campioni di valori del parametro estratti dalla distribuzione a posteriori trasforma un problema di calcolo integrale in un problema di riepilogo dei dati. Abbiamo visto le procedure maggiormente usate che consentono di utilizzare i campioni a posteriori per produrre indici di sintesi della distribuzione a posteriori: gli intervalli di credibilità e le stime puntuali.

Bibliografia

- Gautret, P., J. C. Lagier, P. Parola, L. Meddeb, M. Mailhe, B. Doudier e S. ... Honoré (2020). «Hydroxychloroquine and azithromycin as a treatment of COVID-19: Results of an open-label non-randomized clinical trial». In: *International Journal of Antimicrobial Agents* (cit. alle pp. 2, 9, 11, 15, 16, 18, 21).
- Gelman, A., J. B. Carlin, H. S. Stern e D. B. Rubin (1995). *Bayesian data analysis*. Chapman e Hall/CRC (cit. a p. 2).
- Hulme, O. J., E. J. Wagenmakers, P. Damkier, C. F. Madelung, H. R. Siebner, J. Helweg-Larsen e K. H. ... Madsen (2020). «Reply to Gautret et al. 2020: A Bayesian reanalysis of the effects of hydroxychloroquine and azithromycin on viral carriage in patients with COVID-19». In: *medRxiv* (cit. alle pp. 9, 12).
- Johnson, A. A., M. Ott e M. Dogucu (2022). *Bayes Rules! An Introduction to Bayesian Modeling with R*. CRC Press (cit. alle pp. 14, 19).
- Lee, M. D. e E.-J. Wagenmakers (2014). *Bayesian cognitive modeling: A practical course*. Cambridge university press (cit. a p. 33).
- Zetsche, U., P.-C. Bürkner e B. Renneberg (2019). «Future expectations in clinical depression: Biased or realistic?» In: *Journal of Abnormal Psychology* 128.7, pp. 678–688 (cit. a p. 27).

Elenco delle figure

1.1	Trace-plot per il parametro θ nel modello Beta-Binomiale.	6
1.2	Istogramma che illustra l'approssimazione della distribuzione a posteriori per il parametro θ nel modello Beta-Binomiale.	7
1.3	Istogramma che illustra l'approssimazione della distribuzione a posteriori per il parametro θ nel modello Beta-Binomiale. La curva nera rappresenta la corretta distribuzione a posteriori Beta(16, 4).	8
2.1	Trace plot per il modello Beta-Binomiale dei dati di Gautret et al.(2020).	14
2.2	Trace plots (a sinistra) e corrispondenti grafici di densità (a destra) di due ipotetiche catene di Markov. Queste figure forniscono due esempi di come potrebbero apparire delle catene di Markov non stazionarie. Le linee nere sovrapposte alle densità empiriche (a destra) rappresentano una ipotetica distribuzione target Beta(11,3).	15
2.3	Trace plot (a sinistra) e correlogramma (a destra) di una catena di Markov in cui il mixing è lento – figura riprodotta da @Johnson2022bayesrules.	19

Abstract This document contains the material of the lessons of Psicometria B000286 (2021/2022) aimed at students of the first year of the Degree Course in Psychological Sciences and Techniques of the University of Florence, Italy.

Keywords Data science, Bayesian statistics.