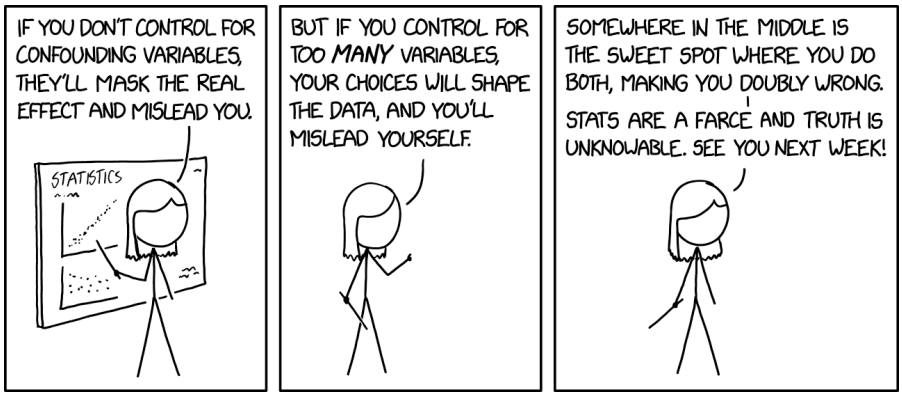
Corrado Caudek

Data Science per psicologi

Psicometria – AA 2021/2022



Indice

Elenco delle figure

Elenco delle tabelle

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. *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 un insieme di 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

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 pon-

te, 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

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

Avendo insegnato molte volte in passato un corso introduttivo di analisi dei dati 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. Istintivamente 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 affontare, 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, opppure 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 possia-

mo 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.[1] È utile farsi una nota mentalmente delle lacune nella vostra comprensione e tornare su di esse in seguito per carcare 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

[1] Ricordatevi inoltre che gli individui tendono a sottostimare la propria capacità di apprendere (Horn and Loewenstein, 2021).

un software, ricordatevi: “Google is your friend”!

Corrado Caudek
Marzo 2022

Parte I

Inferenza bayesiana

1

Flusso di lavoro bayesiano

La moderna statistica bayesiana viene per lo più eseguita utilizzando un linguaggio di programmazione probabilistico implementato su computer. Ciò ha cambiato radicalmente il modo in cui venivano eseguite le statistiche bayesiane anche fin pochi decenni fa. La complessità dei modelli che possiamo costruire è aumentata e la barriera delle competenze matematiche e computazionali che sono richieste è diminuita. Inoltre, il processo di modellazione iterativa è diventato, sotto molti aspetti, molto più facile da eseguire. Anche se formulare modelli statistici complessi è diventato più facile che mai, la statistica è un campo pieno di sottigliezze che non scompaiono magicamente utilizzando potenti metodi computazionali. Pertanto, avere una buona preparazione sugli aspetti teorici, specialmente quelli rilevanti nella pratica, è estremamente utile per applicare efficacemente i metodi statistici.

1.1 Modellizzazione bayesiana

L'analisi bayesiana corrisponde alla costruzione di un modello statistico che si può rappresentare con una quaterna

$$(\mathcal{Y}, p(y \mid \theta), p(\theta), \theta \in \Theta), \tag{1.1}$$

dove $\mathcal{Y}$ è l'insieme di tutti i possibili risultati ottenuti dall'esperimento casuale e $p(y \mid \theta)$ è una famiglia di leggi di probabilità, indicizzata dal parametro $\theta \in \Theta$, che descrive l'incertezza sull'esito dell'esperimento. Secondo l'approccio bayesiano, il parametro incognito θ è considerato una variabile casuale che segue la legge di probabilità $p(\theta)$. L'incertezza su θ è la sintesi delle opinioni e delle informazioni che si hanno sul parametro prima di avere osservato il risultato dell'esperimento e prende il nome di *distribuzione a priori*. La costruzione del modello statistico

passa attraverso la scelta di una densità $p(y \mid \theta)$ che rappresenta, in senso probabilistico, il fenomeno d'interesse, e attraverso la scelta di una distribuzione a priori $p(\theta)$. Le informazioni che si hanno a priori sul parametro di interesse θ, contenute in $p(\theta)$, vengono aggiornate attraverso quelle provenienti dal campione osservato $y = (y_1, \dots, y_n)$ contenute nella funzione $p(y \mid \theta)$, che, osservata come funzione di θ per y, prende il nome di *funzione di verosimiglianza*. L'aggiornamento delle informazioni avviene attraverso la formula di Bayes

$$p(\theta \mid y) = \frac{p(y \mid \theta)p(\theta)}{\int_\Theta p(y \mid \theta)p(\theta)\, \mathrm{d}\theta} \quad \theta \in \Theta, \tag{1.2}$$

in cui $p(\theta \mid y)$ prende il nome di *distribuzione a posteriori*.

Il denominatore del Teorema di Bayes (1.2), che costituisce la costante di normalizzazione, è la densità marginale dei dati (o verosimiglianza marginale). In ambito bayesiano la distribuzione a posteriori viene utilizzata per calcolare le principali quantità di interesse dell'inferenza, ad esempio la media a posteriori di θ.

Possiamo descrivere la modellazione bayesiana distinguendo tre passaggi (Martin et al., 2022).

1. Dati alcuni dati e alcune ipotesi su come questi dati potrebbero essere stati generati, progettiamo un modello combinando e trasformando variabili casuali.
2. Usiamo il teorema di Bayes per condizionare i nostri modelli ai dati disponibili. Chiamiamo questo processo "inferenza" e come risultato otteniamo una distribuzione a posteriori.
3. Critichiamo il modello verificando se il modello abbia senso utilizzando criteri diversi, inclusi i dati e la nostra conoscenza del dominio. Poiché generalmente siamo incerti sui modelli, a volte confrontiamo modelli diversi.

Questi tre passaggi vengono eseguiti in modo iterativo e danno luogo a quello che è chiamato "flusso di lavoro bayesiano" (*bayesian workflow*).

1.1.1 Notazione

Per fissare la notazione, nel seguito y rappresenterà i dati e θ rappresenterà i parametri incogniti di un modello statistico. Sia y che θ ven-

gono concepiti come variabili casuali. Con x vengono invece denotate le quantità note, come ad esempio i predittori del modello lineare. Per rappresentare in un modo conciso i modelli probabilistici viene usata una notazione particolare. Ad esempio, invece di scrivere $p(\theta) = \text{Beta}(1, 1)$ scriviamo $\theta \sim \text{Beta}(1, 1)$. Il simbolo "$\sim$" viene spesso letto "è distribuito come". Possiamo anche pensare che significhi che θ costituisce un campione casuale estratto dalla distribuzione Beta(1, 1). Allo stesso modo, ad esempio, la verosimiglianza del modello binomiale può essere scritta come $y \sim \text{Bin}(n, \theta)$.

1.2 Distribuzioni a priori

Quando adottiamo un approccio bayesiano, i parametri della distribuzione di riferimento non venono considerati come delle costanti incognite ma bensì vengono trattati come variabili casuali; di conseguenza, i parametri assumono una particolare distribuzione che nelle statistica bayesiana viene definita "a priori". I parametri (o il parametro), che possiamo indicare con θ, possono assumere delle distribuzioni a priori differenti: a seconda delle informazioni disponibili bisogna selezionare una distribuzione di θ in modo tale che venga assegnata una probabilità maggiore a quei valori che si ritengono più plausibili per θ. Idealmente, le credenze a priori che portano alla specificazione di una distribuzione a priori dovrebbero essere supportate da una qualche motivazione, come ad esempio i risultati di ricerche precedenti.

1.2.1 Tipologie di distribuzioni a priori

Possiamo distinguere tra diverse distribuzioni a priori in base a quanto fortemente impegnano il ricercatore a ritenere come plausibile un particolare intervallo di valori dei parametri. Il caso più estremo è quello che rivela una totale assenza di conoscenze a priori, il che conduce alle *distribuzioni a priori non informative*, ovvero quelle che assegnano lo stesso livello di credibilità a tutti i valori dei parametri. Le distribuzioni a priori informative, d'altra parte, possono essere *debolmente informative* o *fortemente informative*, a seconda della forza della credenza che esprimono. Il caso più estremo di credenza a priori è quello che riassume il punto di vista del ricercatore nei termini di un *unico valore* del parametro, il che assegna tutta la probabilità (massa o densità) ad di un

singolo valore del parametro. Poiché questa non è più una distribuzione di probabilità, sebbene ne soddisfi la definizione, in questo caso si parla di una *distribuzione a priori degenerata*. La figura seguente motra alcuni esempi di distribuzioni a priori per il modello Binomiale:

- distribuzione *non informativa*: $\theta_c \sim \text{Beta}(1, 1)$;
- distribuzione *debolmente informativa*: $\theta_c \sim \text{Beta}(5, 2)$;
- distribuzione *fortemente informativa*: $\theta_c \sim \text{Beta}(50, 20)$;
- *valore puntuale*: $\theta_c \sim \text{Beta}(\alpha, \beta)$ con $\alpha, \beta \to \infty$ e $\frac{\alpha}{\beta} = \frac{5}{2}$.

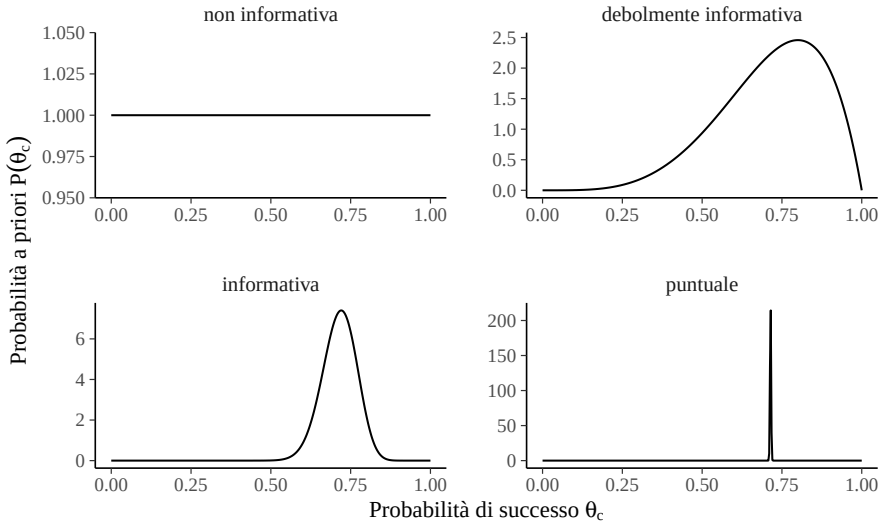


Figura 1.1: *Esempi di distribuzioni a priori per il parametro θ_c nel Modello Binomiale.*

1.2.2 Selezione della distribuzione a priori

La selezione delle distribuzioni a priori è stata spesso vista come una delle scelte più importanti che un ricercatore fa quando implementa un modello bayesiano in quanto può avere un impatto sostanziale sui risultati finali. La soggettività delle distribuzioni a priori è evidenziata dai critici come un potenziale svantaggio dei metodi bayesiani. A questa critica, van de Schoot et al. (2021) rispondono dicendo che, al di là della scelta delle distribuzioni a priori, ci sono molti elementi del processo di inferenza statistica che sono soggettivi, ovvero la scelta del modello statistico e le ipotesi sulla distribuzione degli errori. In secondo luogo, van de Schoot et al. (2021) notano come le distribuzioni a priori svol-

gono due importanti ruoli statistici: quello della "regolarizzazione della stima", ovvero, il processo che porta ad indebolire l'influenza indebita di osservazioni estreme, e quello del miglioramento dell'efficienza della stima, ovvero, la facilitazione dei processi di calcolo numerico di stima della distribuzione a posteriori. L'effetto della distribuzione a priori sulla distribuzione a posteriori verrà discusso in dettaglio nel Capitolo 3.

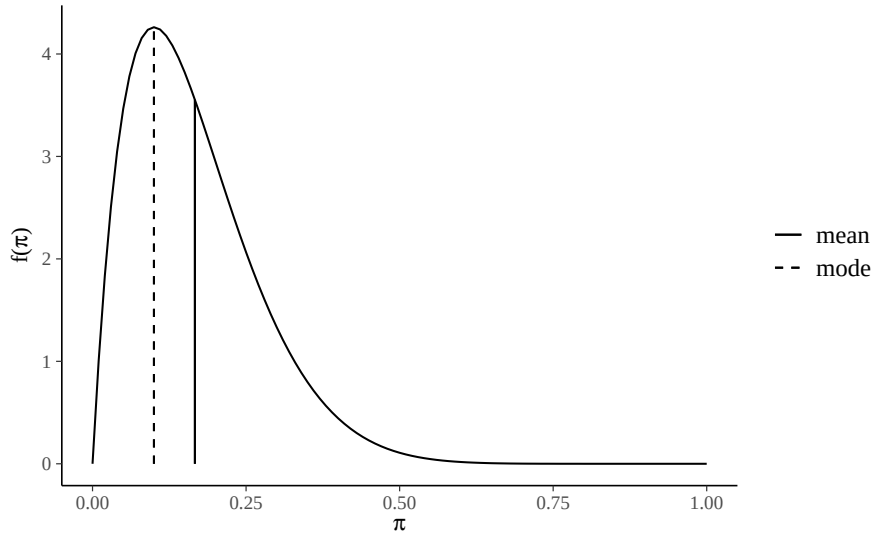
1.2.3 Un esempio concreto

Per introdurre la modelizzazione bayesiana useremo qui i dati riportati da Zetsche et al. (2019) (si veda l'appendice **??**). Tali dati corrispondono a 23 "successi" in 30 prove e possono dunque essere considerati la manifestazione di una variabile casuale Bernoulliana.

Se non abbiamo alcuna informazione a priori su θ (ovvero, la probabilità che l'aspettativa dell'umore futuro del partecipante sia distorta negativamente), potremmo pensare di usare una distribuzione a priori uniforme, ovvero una Beta di parametri $\alpha = 1$ e $\beta = 1$. Una tale scelta, tuttavia, è sconsigliata in quanto è più vantaggioso usare una distribuzione debolmente informativa, come ad esempio Beta$(2, 2)$, che ha come scopo la regolarizzazione, cioè quello di mantenere le inferenze in un intervallo ragionevole. Qui useremo una Beta$(2, 10)$.

$$p(\theta) = \frac{\Gamma(12)}{\Gamma(2)\Gamma(10)}\theta^{2-1}(1-\theta)^{10-1}.$$

```
bayesrules::plot_beta(alpha = 2, beta = 10, mean = TRUE, mode = TRUE)
```



La Beta$(2, 10)$ esprime la credenza che θ assume valori < 0.5, con il valore più plausibile pari a circa 0.1. Questo è assolutamente implausibile per il caso dell'esempio in discussione: la Beta$(2, 10)$ verrà usata solo per scopi didattici, ovvero, per esplorare le conseguenze di tale scelta sulla distribuzione a posteriori.

1.3 La funzione di verosimiglianza

Iniziamo con una definizione.

Definizione 1.1. La *funzione di verosimiglianza* $\mathcal{L}(\theta \mid y) = f(y \mid \theta), \theta \in \Theta$, è la funzione di massa o di densità di probabilità dei dati y vista come una funzione del parametro sconosciuto (o dei parametri sconosciuti) θ.

Detto in altre parole, le funzioni di verosimiglianza e di (massa o densità di) probabilità sono formalmente identiche, ma è completamente diversa la loro interpretazione. Nel caso della funzione di massa o di densità di probabilità la distribuzione del vettore casuale delle osservazioni campionarie y dipende dai valori assunti dal parametro (o dai parametri) θ; nel caso della la funzione di verosimiglianza la credibilità assegnata a ciascun possibile valore θ viene determinata avendo acquisita l'informazione campionaria y che rappresenta l'elemento condizionante. In altri termini, la

funzione di verosimiglianza descrive in termini relativi il sostegno empirico che $\theta \in \Theta$ riceve da y. Infatti, la funzione di verosimiglianza assume forme diverse al variare di y. Possiamo dunque pensare alla funzione di verosimiglianza come alla risposta alla seguente domanda: avendo osservato i dati y, quanto risultano (relativamente) credibili i diversi valori del parametro θ? In termini più formali possiamo dire: sulla base dei dati, $\theta_1 \in \Theta$ risulta più credibile di $\theta_2 \in \Theta$ quale indice del modello probabilistico generatore dei dati se $\mathcal{L}(\theta_1) > \mathcal{L}(\theta_1)$.

Notiamo un punto importante: la funzione $\mathcal{L}(\theta \mid y)$ non è una funzione di densità. Infatti, essa non racchiude un'area unitaria.

1.3.1 Notazione

Seguendo una pratica comune, in questa dispensa spesso useremo la notazione $p(\cdot)$ per rappresentare due quantità differenti, ovvero la funzione di verosimiglianza e la distribuzione a priori. Questo piccolo abuso di notazione riflette il seguente punto di vista: anche se la verosimiglianza non è una funzione di densità di probabilità, noi non vogliamo stressare questo aspetto, ma vogliamo piuttosto pensare alla verosimiglianza e alla distribuzione a priori come a due elementi che sono egualmente necessari per calcolare la distribuzione a posteriori. In altri termini, per così dire, questa notazione assegna lo stesso status epistemologico alle due diverse quantità che si trovano al numeratore della regola di Bayes.

1.3.2 La log-verosimiglianza

Dal punto di vista pratico risulta più conveniente utilizzare, al posto della funzione di verosimiglianza, il suo logaritmo naturale, ovvero la funzione di log-verosimiglianza:

$$\ell(\theta) = \log \mathcal{L}(\theta). \tag{1.3}$$

Poiché il logaritmo è una funzione strettamente crescente (usualmente si considera il logaritmo naturale), allora $\mathcal{L}(\theta)$ e $\ell(\theta)$ assumono il massimo (o i punti di massimo) in corrispondenza degli stessi valori di θ (per un approfondimento, si veda l'Appendice **??**):

$$\widehat{\theta} = \arg\max_{\theta \in \Theta} \ell(\theta) = \arg\max_{\theta \in \Theta} \mathcal{L}(\theta).$$

Per le proprietà del logaritmo, si ha

$$\ell(\theta) = \log \left(\prod_{i=1}^{n} f(y \mid \theta) \right) = \sum_{i=1}^{n} \log f(y \mid \theta). \tag{1.4}$$

Si noti che non è necessario lavorare con i logaritmi, ma è fortemente consigliato. Il motivo è che i valori della verosimiglianza, in cui si moltiplicano valori di probabilità molto piccoli, possono diventare estremamente piccoli – qualcosa come 10^{-34}. In tali circostanze, non è sorprendente che i programmi dei computer mostrino problemi di arrotondamento numerico. Le trasformazioni logaritmiche risolvono questo problema.

1.3.3 Un esempio concreto

Se i dati di Zetsche et al. (2019) possono essere riassunti da una proporzione allora è sensato adottare un modello probabilistico binomiale quale meccanismo generatore dei dati:

$$y \sim \text{Bin}(n, \theta), \tag{1.5}$$

laddove θ è la probabiltà che una prova Bernoulliana assuma il valore 1 e n corrisponde al numero di prove Bernoulliane. Questo modello assume che le prove Bernoulliane y_i che costituiscono il campione y siano tra loro indipendenti e che ciascuna abbia la stessa probabilità $\theta \in [0, 1]$ di essere un "successo" (valore 1). In altre parole, il modello generatore dei dati avrà una funzione di massa di probabilità

$$p(y \mid \theta) \; = \; \text{Bin}(y \mid n, \theta).$$

Nei capitoli precedenti è stato mostrato come, sulla base del modello binomiale, sia possibile assegnare una probabilità a ciascun possibile valore $y \in \{0, 1, \dots, n\}$ *assumendo noto il valore del parametro* θ. Ma ora abbiamo il problema inverso, ovvero quello di fare inferenza su θ alla luce dei dati campionari y. In altre parole, riteniamo di conoscere il modello probabilistico che ha generato i dati, ma di tale modello non conosciamo i parametri: vogliamo dunque ottenere informazioni su θ avendo osservato i dati y.

Per i dati di Zetsche et al. (2019) la funzione di verosimiglianza corrisponde alla funzione binomiale di parametro $\theta \in [0, 1]$ sconosciuto. Abbiamo osservato un "successo" 23 volte in 30 "prove", dunque, $y = 23$ e $n = 30$. La funzione di verosimiglianza diventa

$$\mathcal{L}(\theta \mid y) = \frac{(23+7)!}{23!7!}\theta^{23} + (1-\theta)^7. \tag{1.6}$$

Per costruire la funzione di verosimiglianza dobbiamo applicare la (1.6) tante volte, cambiando ogni volta il valore θ ma *tenendo sempre costante il valore dei dati*. Per esempio, se poniamo $\theta = 0.1$

$$\mathcal{L}(\theta \mid y) = \frac{(23+7)!}{23!7!}0.1^{23} + (1-0.1)^7$$

otteniamo

```
dbinom(23, 30, 0.1)
#> [1] 9.737e-18
```

Se poniamo $\theta = 0.2$

$$\mathcal{L}(\theta \mid y) = \frac{(23+7)!}{23!7!}0.2^{23} + (1-0.2)^7$$

otteniamo

```
dbinom(23, 30, 0.2)
#> [1] 3.581e-11
```

e così via. La figura 1.2 — costruita utilizzando 100 valori equispaziati $\theta \in [0, 1]$ — fornisce una rappresentazione grafica della funzione di verosimiglianza.

```
n <- 30
y <- 23
theta <- seq(0, 1, length.out = 100)
like <- choose(n, y) * theta^y * (1 - theta)^(n - y)
tibble(theta, like) %>%
  ggplot(aes(x = theta, y = like)) +
  geom_line() +
  labs(
    y = expression(L(theta)),
    x = expression("Valori possibili di" ~ theta)
  )
```

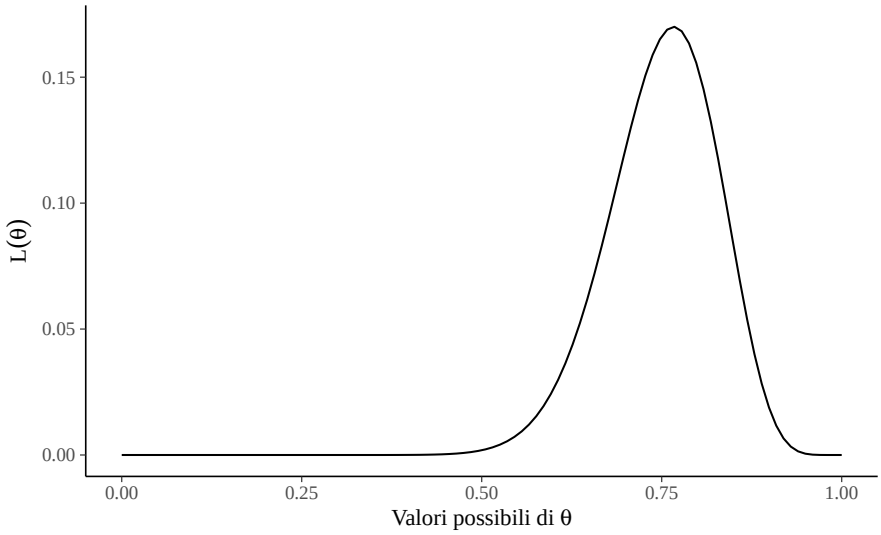


Figura 1.2: *Funzione di verosimiglianza nel caso di 23 successi in 30 prove.*

Come possiamo interpretare la curva che abbiamo ottenuto? Per alcuni valori θ la funzione di verosimiglianza assume valori piccoli; per altri valori θ la funzione di verosimiglianza assume valori più grandi. Questi ultimi sono i valori di θ più credibili e il valore 23/30 (la moda della funzione di verosimiglianza) è il valore più credibile di tutti.

1.4 La verosimiglianza marginale

Per il calcolo di $p(\theta \mid y)$ è necessario dividere il prodotto tra la distribuzione a priori e la verosimiglianza per una costante di normalizzazione. Tale costante di normalizzazione, detta *verosimiglianza marginale*, ha lo scopo di fare in modo che $p(\theta \mid y)$ abbia area unitaria.

Si noti che, nel caso di variabili continue, la verosimiglianza marginale è espressa nei termini di un integrale. Tranne in pochi casi particolari, tale integrale non ha una soluzione analitica. Per questa ragione, l'inferenza bayesiana procede calcolando una approssimazione della distribuzione a posteriori mediante metodi numerici.

1.4.1 Un esempio concreto

Consideriamo nuovamente i dati di Zetsche et al. (2019). Supponiamo che nel numeratore bayesiano la verosimiglianza sia moltiplicata per una distribuzione uniforme, ovvero Beta(1, 1). In tali circostanze, il prodotto si riduce alla funzione di verosimiglianza. Per i dati di Zetsche et al. (2019), dunque, la costante di normalizzazione si ottiene marginalizzando la funzione di verosimiglianza $p(y = 23, n = 30 \mid \theta)$ sopra θ, ovvero risolvendo l'integrale:

$$p(y = 23, n = 30) = \int_0^1 \binom{30}{23} \theta^{23}(1-\theta)^7 \, \mathrm{d}\theta. \tag{1.7}$$

Una soluzione numerica si trova facilmente usando R:

```
like_bin <- function(theta) {
  choose(30, 23) * theta^23 * (1 - theta)^7
}
integrate(like_bin, lower = 0, upper = 1)$value
#> [1] 0.03226
```

La derivazione analitica è fornita nell'Appendice **??**.

1.5 Distribuzione a posteriori

La distribuzione a postreriori si trova applicando il teorema di Bayes:

$$\text{probabilità a posteriori} = \frac{\text{probabilità a priori} \cdot \text{verosimiglianza}}{\text{costante di normalizzazione}}$$

Una volta trovata la distribuzione a posteriori, possiamo usarla per derivare altre quantità di interesse. Questo viene generalmente ottenuto calcolando il valore atteso:

$$J = \int f(\theta) p(\theta \mid y) \, \mathrm{d}y$$

Se $f(\cdot)$ è la funzione identità, ad esempio, J risulta essere la media di θ:

$$\bar{\theta} = \int_{\Theta} \theta p(\theta \mid y) \, \mathrm{d}\theta.$$

Ripeto qui quanto detto sopra: le quantità di interesse della statistica bayesiana(costante di normalizzazione, valore atteso della distribuzione a posteriori, ecc.) contengono integrali che risultano, nella maggior parte dei casi, impossibili da risolvere analiticamente. Per questo motivo, si ricorre a metodi di stima numerici, in particolare a quei metodi Monte Carlo basati sulle proprietà delle catene di Markov (MCMC). Questo argomento verrà discusso nel Capitolo 4.

1.6 Distribuzione predittiva a priori

La distribuzione a posteriori è l'oggetto centrale nella statistica bayesiana, ma non è l'unico. Oltre a fare inferenze sui valori dei parametri, potremmo voler fare inferenze sui dati. Questo può essere fatto calcolando la *distribuzione predittiva a priori*:

$$p(y^*) = \int_{\Theta} p(y^* \mid \theta) p(\theta) \, \mathrm{d}\theta. \tag{1.8}$$

La (1.8) descrive la distribuzione prevista dei dati in base al modello (che include la distribuzione a priori e la verosimiglianza), ovvero descrive i dati y^* che ci aspettiamo di osservare, dato il modello, prima di avere osservato i dati del campione.

È possibile utilizzare campioni dalla distribuzione predittiva a priori per valutare e calibrare i modelli utilizzando le nostre conoscenze dominio-specifiche. Ad esempio, ci possiamo chiedere: "È sensato che un modello dell'altezza umana preveda che un essere umano sia alto -1.5 metri?". Già prima di misurare una singola persona, possiamo renderci conto dell'assurdità di questa domanda. Se la distribuzione prevista dei dati consente domande di questo tipo (ovvero, prevede di osservare dati che risultano insensati alla luce delle nostre conoscenze dominio-specifiche), è chiaro che il modello deve essere riformulato.

1.7 Distribuzione predittiva a posteriori

Un'altra quantità utile da calcolare è la distribuzione predittiva a posteriori:

$$p(\tilde{y} \mid y) = \int_{\Theta} p(\tilde{y} \mid \theta) p(\theta \mid y) \, \mathrm{d}\theta. \tag{1.9}$$

Questa è la distribuzione dei dati attesi futuri $\tilde{y}$ alla luce della distribuzione a posteriori $p(\theta \mid y)$, che a sua volta è una conseguenza del modello adottato (distribuzione a priori e verosimiglianza) e dei dati osservati. In altre parole, questi sono i dati che il modello si aspetta dopo aver osservato i dati de campione. Dalla (1.9) possiamo vedere che le previsioni sui dati attesi futuri sono calcolate integrando (o marginalizzando) sulla distribuzione a posteriori dei parametri. Di conseguenza, le previsioni calcolate in questo modo incorporano l'incertezza relativa alla stima dei parametri del modello.

Commenti e considerazioni finali

Questo Capitolo ha brevemente passato in rassegna i concetti di base dell'inferenza statistica bayesiana. In base all'approccio bayesiano, invece di dire che il parametro di interesse di un modello statistico ha un valore vero ma sconosciuto, diciamo che, prima di eseguire l'esperimento, è possibile assegnare una distribuzione di probabilità, che chiamano stato di credenza, a quello che è il vero valore del parametro. Questa distribuzione a priori può essere nota (per esempio, sappiamo che la distribuzione dei punteggi del QI è normale con media 100 e deviazione standard 15) o può essere del tutto arbitraria. L'inferenza bayesiana procede poi nel modo seguente: si raccolgono alcuni dati e si calcola la probabilità dei possibili valori del parametro alla luce dei dati osservati e delle credenze a priori. Questa nuova distribuzione di probabilità è chiamata "distribuzione a posteriori" e riassume l'incertezza dell'inferenza.

2

Distribuzioni a priori coniugate

Obiettivo di questo Capitolo è fornire un esempio di derivazione della distribuzione a posteriori scegliendo quale distribuzione a priori una distribuzione coniugata. Esamineremo qui il lo schema beta-binomiale.

2.1 Lo schema beta-binomiale

Iniziamo con una definizione.

Definizione 2.1. Una distribuzione di probabilità a priori $p(\theta)$ si dice *coniugata* al modello usato se la distribuzione a priori e la distribuzione a posteriori hanno la stessa forma funzionale. Dunque, le due distribuzioni differiscono solo per il valore dei parametri.

Ad esempio, se la distribuzione a priori è una distribuzione Beta e se la funzione di verosimiglianza è binomiale, allora anche la distribuzione a posteriori sarà una distribuzione Beta.

Da un punto di vista matematico, le distribuzioni a priori coniugate sono la scelta più conveniente in quanto ci consentono di calcolare analiticamente la distribuzione a posteriori con "carta e penna", senza la necessità di ricorrere a calcoli complessi. Da una prospettiva computazionale moderna, però, le distribuzioni a priori coniugate generalmente non sono migliori delle alternative, dato che i moderni metodi computazionali ci consentono di eseguire l'inferenza praticamente con qualsiasi scelta delle distribuzioni a priori, e non solo con le distribuzioni a priori che risultano matematicamente convenienti. Tuttavia, le famiglie coniugate offrono un utile ausilio didattico nello studio dell'inferenza bayesiana. Questo è il motivo per cui le esamineremo qui. Nello specifico, esamineremo quello che viene chiamato lo schema beta-binomiale.

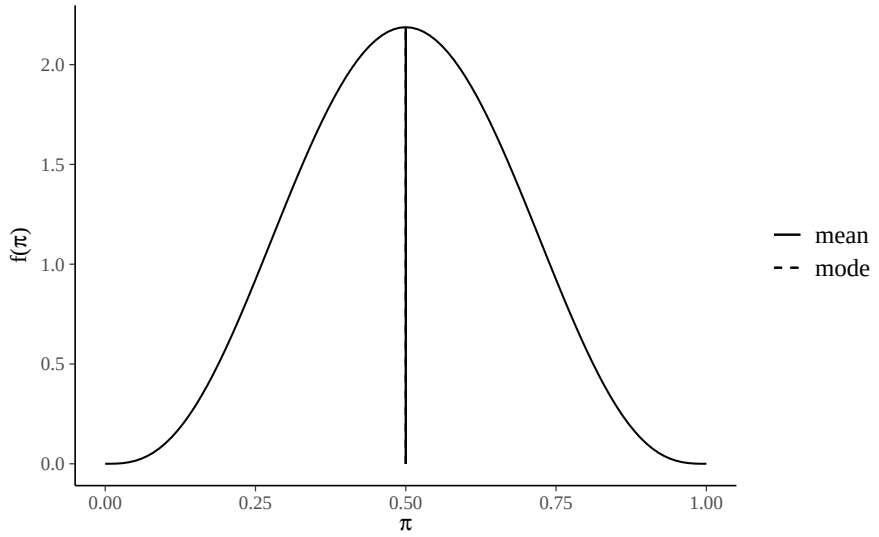
Per fare un esempio concreto, consideriamo nuovamente i dati di Zetsche et al. (2019): nel campione di 30 partecipanti clinici le aspettative future di 23 partecipanti risultano negativamente distorte mentre quelle di 7 partecipanti risultano positivamente distorte. Nel seguito, indicheremo con θ la probabilità che le aspettative di un paziente clinico siano distorte negativamente. Ci poniamo il problema di ottenere una stima a posteriori di θ avendo osservato 23 "successi" in 30 prove.

I dati osservati ($y = 23$) possono essere considerati la manifestazione di una variabile casuale Bernoulliana. In tali circostanze, esiste una famiglia di distribuzioni che, qualora venga scelta per la distribuzione a priori, fa sì che la distribuzione a posteriori abbia la stessa forma funzionale della distribuzione a priori. Questo consente una soluzione analitica dell'integrale che compare a denominatore nella formula di Bayes. Nel caso presente, la famiglia di distribuzioni che ha questa proprietà è la distribuzione Beta.

2.1.1 La specificazione della distribuzione a priori

È possibile esprimere diverse credenze iniziali rispetto a θ mediante la distribuzione Beta. Ad esempio, la scelta di una Beta($\alpha = 4, \beta = 4$) quale distribuzione a priori per il parametro θ corrisponde alla credenza a priori che associa all'evento "presenza di una aspettativa futura distorta negativamente" una grande incertezza: il valore 0.5 è il valore di θ più plausibile, ma anche gli altri valori del parametro (tranne gli estremi) sono ritenuti piuttosto plausibili. Questa distribuzione a priori esprime la credenza che sia egualmente probabile per un'aspettativa futura essere distorta negativamente o positivamente.

```
library("bayesrules")
plot_beta(alpha = 4, beta = 4, mean = TRUE, mode = TRUE)
```

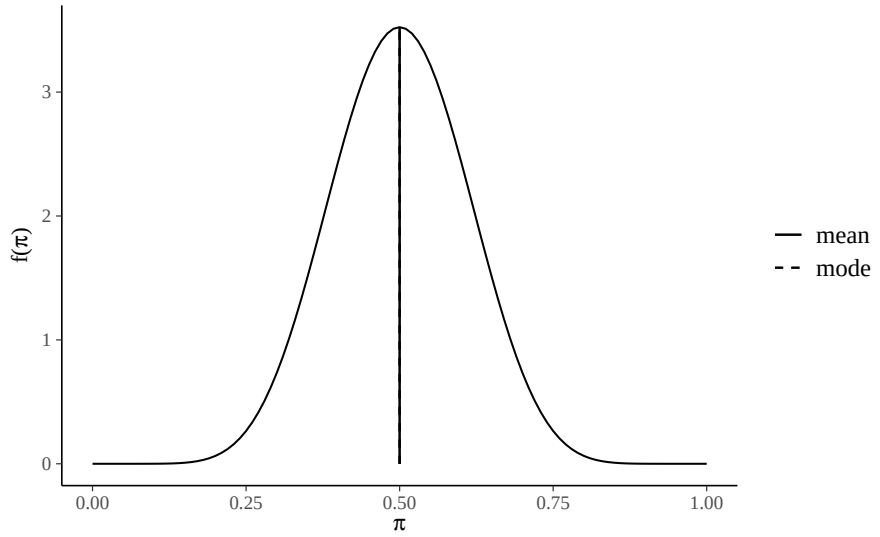


Possiamo quantificare la nostra incertezza calcolando, con un grado di fiducia del 95%, la regione nella quale, in base a tale credenza a priori, si trova il valore del parametro. Per ottenere tale intervallo di credibilità a priori, usiamo la funzione `qbeta()` di R. In `qbeta()` i parametri α e β sono chiamati `shape1` e `shape2`:

```
qbeta(c(0.025, 0.975), shape1 = 4, shape2 = 4)
#> [1] 0.1841 0.8159
```

Se poniamo $\alpha = 10$ e $\beta = 10$, questo corrisponde ad una credenza a priori che sia egualmente probabile per un'aspettativa futura essere distorta negativamente o positivamente,

```
plot_beta(alpha = 10, beta = 10, mean = TRUE, mode = TRUE)
```



ma ora la nostra certezza a priori sul valore del parametro è maggiore, come indicato dall'intervallo al 95%:

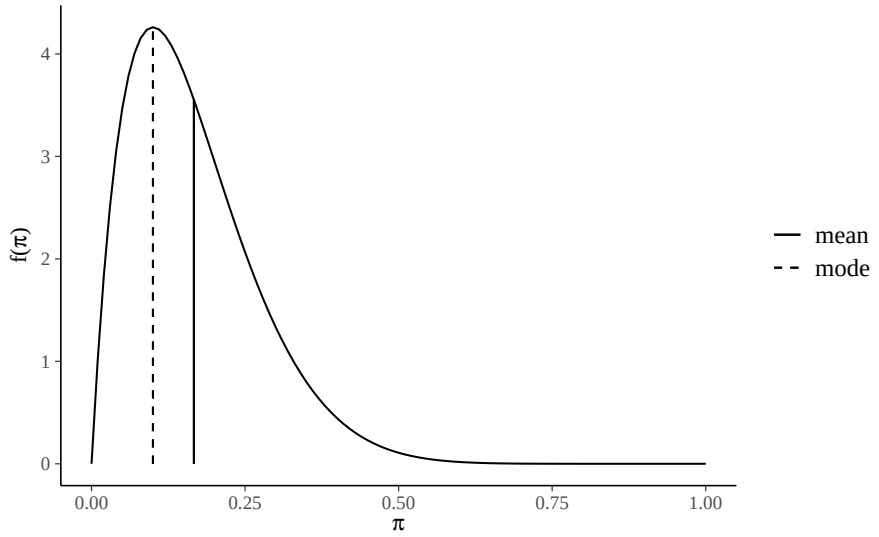
```
qbeta(c(0.025, 0.975), shape1 = 10, shape2 = 10)
#> [1] 0.2886 0.7114
```

Quale distribuzione a priori dobbiamo scegliere? In un problema concreto di analisi dei dati, la scelta della distribuzione a priori dipende dalle credenze a priori che vogliamo includere nell'analisi dei dati. Se non abbiamo alcuna informazione a priori, potremmo usare $\alpha = 1$ e $\beta = 1$, che produce una distribuzione a priori uniforme. Ma l'uso di distribuzioni a priori uniformi è sconsigliato per vari motivi, inclusa l'instabilità numerica della stima dei parametri. È meglio invece usare una distribuzione a priori debolmente informativa, come Beta$(2, 2)$.

Nella discussione presente, solo per fare un esempio, useremo quale distribuzione a priori una Beta$(2, 10)$, ovvero:

$$p(\theta) = \frac{\Gamma(12)}{\Gamma(2)\Gamma(10)}\theta^{2-1}(1-\theta)^{10-1}.$$

```
plot_beta(alpha = 2, beta = 10, mean = TRUE, mode = TRUE)
```



La Beta$(2, 10)$ esprime la credenza che $\theta < 0.5$, con il valore più plausibile pari a cicrca 0.1.

2.1.2 La specificazione della distribuzione a posteriori

Una volta scelta una distribuzione a priori di tipo Beta, i cui parametri rispecchiano le nostre credenze iniziali su θ, la distribuzione a posteriori viene specificata dalla formula di Bayes:

$$\text{distribuzione a posteriori} = \frac{\text{verosimiglianza} \cdot \text{distribuzione a priori}}{\text{verosimiglianza marginale}}.$$

Nel caso presente abbiamo

$$p(\theta \mid n = 30, y = 23) = \frac{\left[\binom{30}{23}\theta^{23}(1-\theta)^{30-23}\right]\left[\frac{\Gamma(12)}{\Gamma(2)\Gamma(10)}\theta^{2-1}(1-\theta)^{10-1}\right]}{p(y = 23)},$$

laddove $p(y = 23)$, ovvero la verosimiglianza marginale, è una costante di normalizzazione.

Riscriviamo l'equazione precedente in termini più generali:

$$p(\theta \mid n, y) = \frac{\left[\binom{n}{y}\theta^{y}(1-\theta)^{n-y}\right]\left[\frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)}\theta^{a-1}(1-\theta)^{b-1}\right]}{p(y)}$$

Raccogliendo tutte le costanti otteniamo:

$$p(\theta \mid n, y) = \left[\frac{\binom{n}{y} \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)}}{p(y)} \right] \theta^y (1-\theta)^{n-y} \theta^{a-1} (1-\theta)^{b-1}.$$

Se ignoriamo il termine costante all'interno della parentesi quadra

$$\begin{aligned} p(\theta \mid n, y) &\propto \theta^y (1-\theta)^{n-y} \theta^{a-1} (1-\theta)^{b-1}, \\ &\propto \theta^{a+y-1} (1-\theta)^{b+n-y-1}, \end{aligned}$$

il termine di destra dell'equazione precedente identifica il *kernel* della distribuzione a posteriori e corrisponde ad una Beta *non normalizzata* di parametri $a + y$ e $b + n - y$.

Per ottenere una distribuzione di densità, dobbiamo aggiungere una costante di normalizzazione al kernel della distribuzione a posteriori. In base alla definizione della distribuzione Beta, ed essendo $a' = a + y$ e $b' = b + n - y$, tale costante di normalizzazione sarà uguale a

$$\frac{\Gamma(a' + b')}{\Gamma(a')\Gamma(b')} = \frac{\Gamma(a + b + n)}{\Gamma(a + y)\Gamma(b + n - y)}.$$

In altri termini, nel caso dello schema beta-binomiale, la distribuzione a posteriori è una $\text{Beta}(a + y, b + n - y)$:

$$\text{Beta}(a + y, b + n - y) = \frac{\Gamma(a + b + n)}{\Gamma(a + y)\Gamma(b + n - y)} \theta^{a+y-1} (1-\theta)^{b+n-y-1}.$$

In sintesi, moltiplicando verosimiglianza $\text{Bin}(n = 30, y = 23 \mid \theta)$ per la la distribuzione a priori $\theta \sim \text{Beta}(2, 10)$ e dividendo per la costante di normalizzazione, abbiamo ottenuto la distribuzione a posteriori $p(\theta \mid n, y) \sim \text{Beta}(25, 17)$. Questo è un esempio di analisi coniugata. La presente combinazione di verosimiglianza e distribuzione a priori è chiamata caso coniugato *beta-binomiale* ed è descritta dal seguente teorema.

Teorema 2.1. *Sia data la funzione di verosimiglianza* $Bin(n, y \mid \theta)$ *e sia* $Beta(\alpha, \beta)$ *una distribuzione a priori. In tali circostanze, la distribuzione a posteriori del parametro* θ *sarà una distribuzione* $Beta(\alpha + y, \beta + n - y)$.

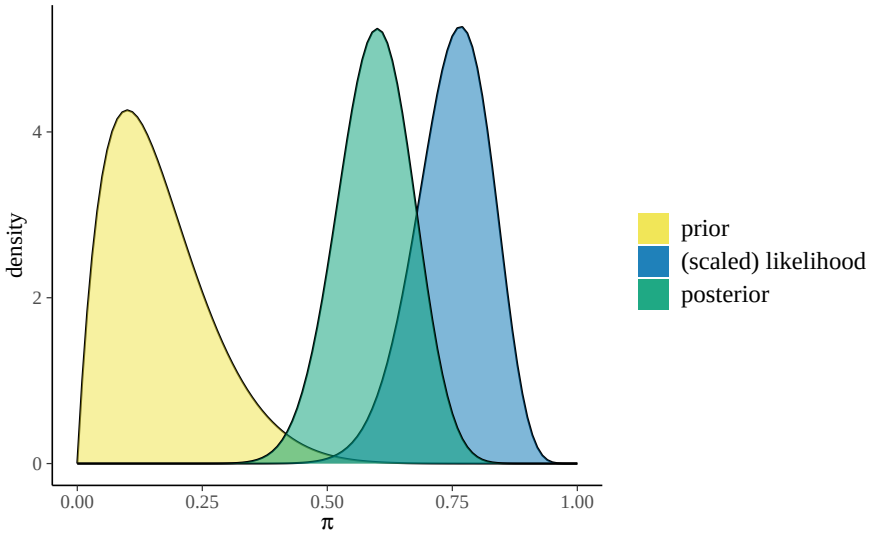
È facile calcolare il valore atteso a posteriori di θ. Essendo $\mathbb{E}[\text{Beta}(\alpha, \beta)] = \frac{\alpha}{\alpha+\beta}$, il risultato cercato diventa

$$\mathbb{E}_{\text{post}}[\text{Beta}(\alpha + y, \beta + n - y)] = \frac{\alpha + y}{\alpha + \beta + n}. \tag{2.1}$$

Esercizio 2.1. Si rappresenti in maniera grafica e si descriva in forma numerica l'aggiornamento bayesiano beta-binomiale per i dati di Zetsche et al. (2019). Si assuma una distribuzione a priori Beta$(2, 10)$.

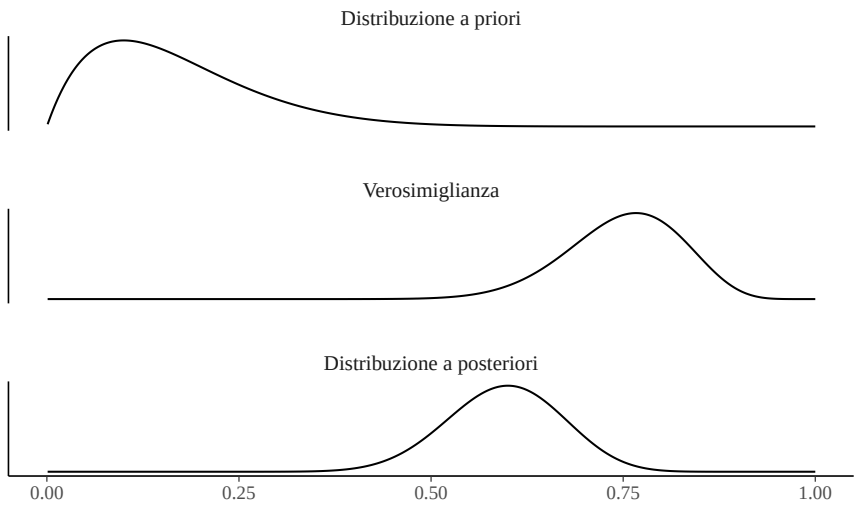
Per i dati in questione, l'aggiornamento bayesiano può essere rappresentato in forma grafica usando la funzione `plot_beta_binomial()` del pacchetto `bayesrules`:

```
bayesrules::plot_beta_binomial(
  alpha = 2, beta = 10, y = 23, n = 30
  )
```



Oppure, possiamo scrivere noi stessi una funzione, come ad esempio la funzione `plot_beta_binom()` riportata in Appendice **??**. Mediante tale la funzione otteniamo

```
plot_beta_bin(2, 10, 23, 30)
```



Un sommario delle distribuzioni a priori e a posteriori può essere ottenuto, ad esempio, usando la funzione `summarize_beta_binomial()` del pacchetto `bayesrules`:

```
bayesrules:::summarize_beta_binomial(
  alpha = 2, beta = 10, y = 23, n = 30
)
#>       model alpha beta   mean mode      var      sd
#> 1     prior     2   10 0.1667  0.1 0.010684 0.10336
#> 2 posterior    25   17 0.5952  0.6 0.005603 0.07485
```

Esercizio 2.2. Per i dati di Zetsche et al. (2019), si trovino la media, la moda, la deviazione standard della distribuzione a posteriori di θ. Si trovi inoltre l'intervallo di credibilità a posteriori del 95% per il parametro θ.

Usando il Teorema 2.1, l'intervallo di credibilità a posteriori del 95% per il parametro θ è:

```
qbeta(c(0.025, 0.975), shape1 = 25, shape2 = 17)
#> [1] 0.4450 0.7368
```

Usando la (2.1), la media della distribuzione a posteriori è

```
25 / (25 + 17)
#> [1] 0.5952
```

Per le proprietà della distribuzione Beta, la moda della distribuzione a posteriori è

```
(25 - 1) / (25 + 17 - 2)
#> [1] 0.6
```

e la deviazione standard della distribuzione a priori è

```
sqrt((25 * 17) / ((25 + 17)^2 * (25 + 17 + 1)))
#> [1] 0.07485
```

Esercizio 2.3. Si trovino i parametri e le proprietà della distribuzione a posteriori del parametro θ per i dati dell'esempio relativo alla ricerca di Stanley Milgram discussa da Johnson et al. (2022).

Nel 1963, Stanley Milgram presentò una ricerca sulla propensione delle persone a obbedire agli ordini di figure di autorità, anche quando tali ordini possono danneggiare altre persone (Milgram, 1963). Nell'articolo, Milgram descrive lo studio come *"consist[ing] of ordering a naive subject to administer electric shock to a victim. A simulated shock generator is used, with 30 clearly marked voltage levels that range from IS to 450 volts. The instrument bears verbal designations that range from Slight Shock to Danger: Severe Shock. The responses of the victim, who is a trained confederate of the experimenter, are standardized. The orders to administer shocks are given to the naive subject in the context of a 'learning experiment' ostensibly set up to study the effects of punishment on memory. As the experiment proceeds the naive subject is commanded to administer increasingly more intense shocks to the victim, even to the point of reaching the level marked Danger: Severe Shock."*

All'insaputa del partecipante, gli shock elettrici erano falsi e l'attore stava solo fingendo di provare il dolore dello shock.

Johnson et al. (2022) fanno inferenza sui risultati dello studio di Milgram mediante il modello Beta-Binomiale. Il parametro di interesse è θ, la probabiltà che una persona obbedisca all'autorità (in questo caso,

somministrando lo shock più severo), anche se ciò significa recare danno ad altri. Johnson et al. (2022) ipotizzano che, prima di raccogliere dati, le credenze di Milgram relative a θ possano essere rappresentate mediante una Beta$(1, 10)$. Sia $y = 26$ il numero di soggetti che, sui 40 partecipanti allo studio, aveva accettato di infliggere lo shock più severo. Assumendo che ogni partecipante si comporti indipendentemente dagli altri, possiamo modellare la dipendenza di y da θ usando la distribuzione binomiale. Giungiamo dunque al seguente modello bayesiano Beta-Binomiale:

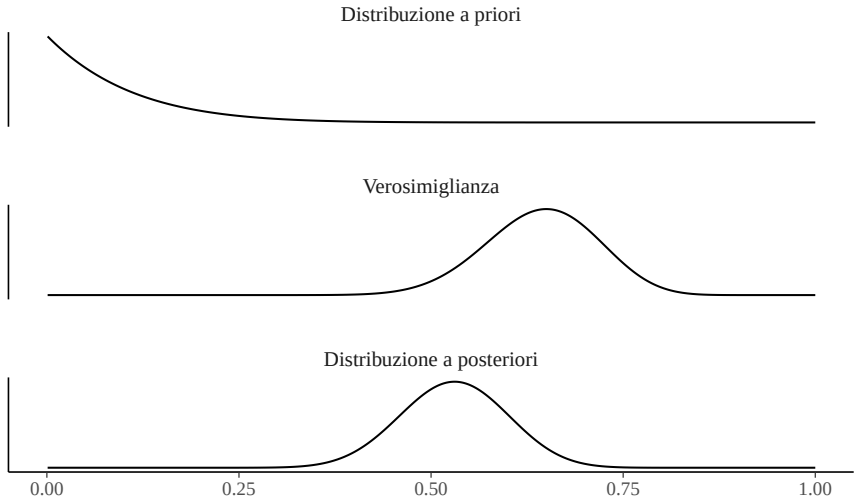
$$\begin{aligned} y \mid \theta &\sim \text{Bin}(n = 40, \theta) \\ \theta &\sim \text{Beta}(1, 10) \ . \end{aligned}$$

Usando le funzioni di `bayesrules` possiamo facilmente calcolare i parametri e le proprietà della distribuzione a posteriori:

```
bayesrules:::summarize_beta_binomial(
  alpha = 1, beta = 10, y = 26, n = 40
)
#>       model alpha beta    mean   mode      var      sd
#> 1     prior     1   10 0.09091 0.0000 0.006887 0.08299
#> 2 posterior    27   24 0.52941 0.5306 0.004791 0.06922
```

Il processo di aggiornamento bayesiano è descritto dalla figura seguente:

```
plot_beta_bin(1, 10, 26, 40)
```



2.2 Principali distribuzioni coniugate

Esistono molte altre combinazioni simili di verosimiglianza e distribuzione a priori le quali producono una distribuzione a posteriori che ha la stessa densità della distribuzione a priori. Sono elencate qui sotto le più note coniugazioni tra modelli statistici e distribuzioni a priori.

- Per il modello Normale-Normale $\mathcal{N}(\mu, \sigma_0^2)$, la distribizione iniziale è $\mathcal{N}(\mu_0, \tau^2)$ e la distribuzione finale è $\mathcal{N}\left(\frac{\mu_0\sigma^2+\bar{y}n\tau^2}{\sigma^2+n\tau^2}, \frac{\sigma^2\tau^2}{\sigma^2+n\tau^2}\right)$.
- Per il modello Poisson-gamma $\text{Po}(\theta)$, la distribizione iniziale è $\Gamma(\lambda, \delta)$ e la distribuzione finale è $\Gamma(\lambda + n\bar{y}, \delta + n)$.
- Per il modello esponenziale $\text{Exp}(\theta)$, la distribizione iniziale è $\Gamma(\lambda, \delta)$ e la distribuzione finale è $\Gamma(\lambda + n, \delta + n\bar{y})$.
- Per il modello uniforme-Pareto $\text{U}(0, \theta)$, la distribizione iniziale è $\text{Pa}(\alpha, \varepsilon)$ e la distribuzione finale è $\text{Pa}(\alpha + n, \max(y_{(n)}, \varepsilon))$.

Commenti e considerazioni finali

Lo scopo di questa discussione è mostrare come sia possibile combinare le nostre conoscenze a priori (espresse nei termini di una densità di probabilità) con le evidenze fornite dai dati (espresse nei termini della funzione di verosimiglianza), così da determinare, mediante il teorema di Bayes, una distribuzione a posteriori, la quale condensa l'incertezza che abbiamo sul parametro θ. Per illustrare tale problema, abbiamo considerato una situazione nella quale θ corrisponde alla probabilità di successo in una sequenza di prove Bernoulliane. In tali circostanze è ragionevole esprimere le nostre credenze a priori mediante la densità Beta, con opportuni parametri. L'inferenza rispetto a θ può essere svolta utilizzando una distribuzione a priori Beta e una verosimiglianza binomiale. Così facendo, la distribuzione a posteriori diventa essa stessa una distribuzione Beta – questo è il cosiddetto schema beta-binomiale. Dato che utilizza una distribuzione a priori coniugata, lo schema beta-binomiale rende possibile la determinazione analitica dei parametri della distribuzione a posteriori.

3

L'influenza della distribuzione a priori

La notazione $p(\theta \mid y) \propto p(\theta)\ p(y \mid \theta)$ rende particolarmente chiaro che la distribuzione a posteriori è un "miscuglio" della distribuzione a priori e della verosimiglianza. Prima di preoccuparci di come calcolare la distribuzione a posteriori, cerchiamo di capire meglio cosa significa "mescolare" la distribuzione a priori e la verosimiglianza. Considereremo qui un esempio discusso da Johnson et al. (2022).

3.1 Il test di Benchdel

Nel fumetto di Alison Bechdel *The Rule*, un personaggio afferma di guardare un film solo se soddisfa le seguenti tre regole (Bechdel, 1986):

- almeno due caratteri nel film devono essere donne;
- queste due donne si parlano;
- parlano di qualcosa altro oltre a parlare di qualche uomo.

Questi criteri costituiscono il *test di Bechdel* per la rappresentazione delle donne nei film. Johnson et al. (2022) pongono la seguente domanda "Quale percentuale dei film che avete visto supera il test di Bechdel?".

Sia $\pi \in [0, 1]$ una variabile casuale che indica la proporzione sconosciuta di film che superano il test di Bechdel. Tre amiche — la femminista, l'ignara e l'ottimista — hanno opionioni diverse su π. Riflettendo sui film che ha visto, la femminista capisce che nella maggioranza dei film mancano personaggi femminili forti. L'ignara non ricorda bene i film che ha visto, quindi non sa quanti film superano il test di Bechdel. Infine, l'ottimista pensa che, in generale, le donne sono ben rappresentate all'interno dei film: secondo lei quasi tutti i film superano il test di Bechdel. Le tre amiche hanno dunque tre modelli a priori diversi di π.

Abbiamo visto in precedenza come sia possibile usare la distribuzione Beta per rappresentare le credenze a priori. Ponendo la gran parte della massa della probabilità a priori su valori $\pi < 0.5$, la distribuzione a priori Beta$(5, 11)$ riflette il punto di vista femminista secondo il quale la maggioranza dei film non supera il test di Bechdel. Al contrario, la Beta$(14, 1)$ pone la gran parte della massa della distribuzione a priori su valori π prossimi a 1, e corrisponde quindi alle credenze a priori dell'amica ottimista. Infine, una Beta$(1, 1)$ o $Unif(0, 1)$, assegna lo stesso livello di plausibilità a tutti i valori $\pi \in [0, 1]$, e corrisponde all'incertezza a priori dell'ignara.

Nell'esempio di Johnson et al. (2022), le tre amiche decidono di rivedere un campione di n film e di registrare y, ovvero il numero di film che superano il test di Bechdel. Se y corrisponde al numero di "successi" in un numero fisso di n prove Bernoulliane i.i.d., allora la dipendenza di y da π viene specificata nei termini di un modello binomiale. Quindi, per ciascuna delle tre amiche è possibile scrivere un modello beta-binomiale

$$\begin{aligned} Y \mid \pi &\sim \text{Bin}(n, \pi) \\ \pi &\sim \text{Beta}(\alpha, \beta) \end{aligned}$$

che utilizza diversi parametri α e β per la distribuzione a priori e che conduce a tre diverse distribuzioni a posteriori per il parametro sconosciuto π:

$$\pi \mid (Y = y) \sim \text{Beta}(\alpha + y, \beta + n - y). \tag{3.1}$$

Johnson et al. (2022) si chiedono come le credenze a priori delle tre amiche influenzano le conclusioni a posteriori a cui esse giungono, dopo avere osservato i dati. Si chiedono inoltre in che modo la dimensione del campione moduli l'influenza della distribuzione a priori sulla distribuzione a posteriori. Per rispondere a queste domande, Johnson et al. (2022) consideriamo tre diversi scenari:

- gli stessi dati osservati, ma distribuzioni a priori diverse;
- dati diversi, ma la stessa distribuzione a priori;
- dati diversi e distribuzioni a priori diverse.

3.2 Stessi dati ma diverse distribuzioni a priori

Iniziamo con lo scenario che descrive il caso in cui abbiamo gli stessi dati ma diverse distribuzioni a priori. Supponiamo che le tre amiche decidano di guardare insieme 20 film selezionati a caso:

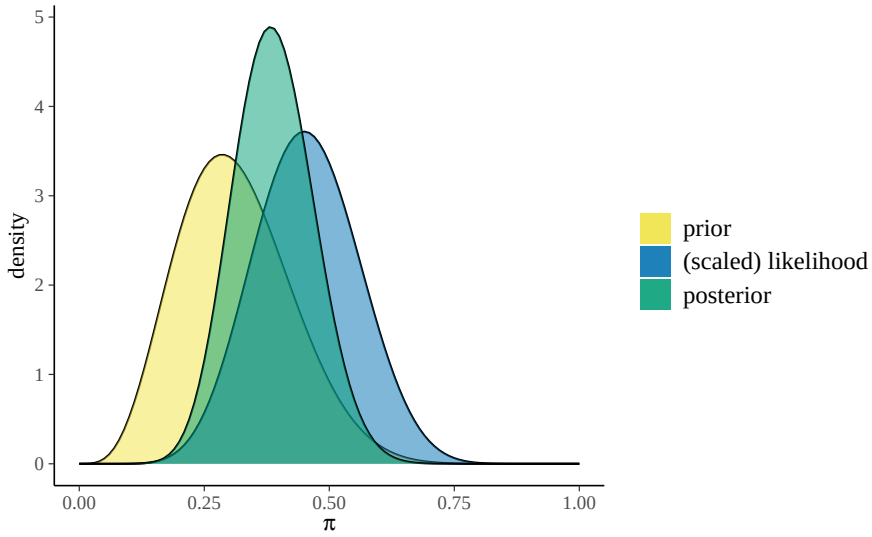
```
data(bechdel, package = "bayesrules")
set.seed(84735)
bechdel_20 <- bechdel %>%
  sample_n(20)
bechdel_20 %>%
  head(3)
#> # A tibble: 3 x 3
#>    year title      binary
#>   <dbl> <chr>      <chr>
#> 1  2005 King Kong  FAIL
#> 2  1983 Flashdance PASS
#> 3  2013 The Purge  FAIL
```

Di questi 20 film, solo il 45% ($y = 9$) passa il test di Bechdel:

```
bechdel_20 %>%
  janitor::tabyl(binary) %>%
  janitor::adorn_totals("row")
#>  binary  n percent
#>    FAIL 11    0.55
#>    PASS  9    0.45
#>   Total 20    1.00
```

Esaminiamo ora le tre distribuzioni a posteriori. Per la femminista abbiamo:

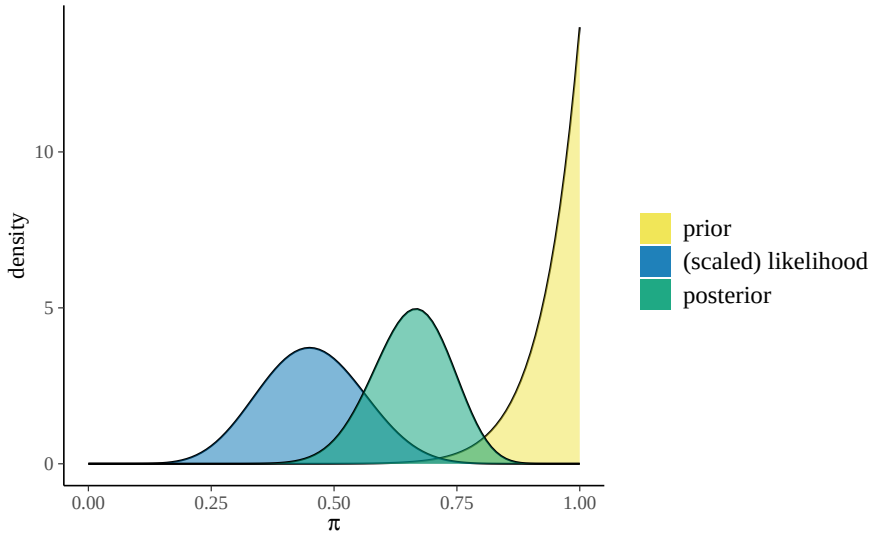
```
bayesrules:::plot_beta_binomial(
  alpha = 5, beta = 11, y = 9, n = 20
  )
```




```
bayesrules:::summarize_beta_binomial(
  alpha = 5, beta = 11, y = 9, n = 20
)
#>       model alpha beta   mean   mode      var      sd
#> 1     prior     5   11 0.3125 0.2857 0.012638 0.11242
#> 2 posterior    14   22 0.3889 0.3824 0.006423 0.08014
```

Per l'ottimista abbiamo:

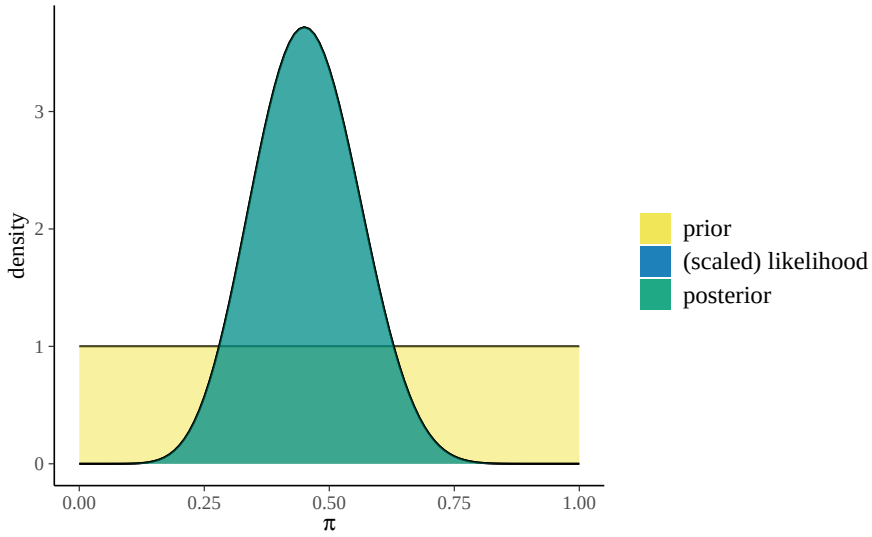
```
bayesrules:::plot_beta_binomial(
  alpha = 14, beta = 1, y = 9, n = 20
)
```




```
bayesrules:::summarize_beta_binomial(
  alpha = 14, beta = 1, y = 9, n = 20
)
#>       model alpha beta   mean   mode      var      sd
#> 1     prior    14    1 0.9333 1.0000 0.003889 0.06236
#> 2 posterior    23   12 0.6571 0.6667 0.006259 0.07911
```

Infine, per l'ignara troviamo

```
bayesrules:::plot_beta_binomial(
  alpha = 1, beta = 1, y = 9, n = 20
)
```




```
bayesrules:::summarize_beta_binomial(
  alpha = 1, beta = 1, y = 9, n = 20
)
#>       model alpha beta   mean mode     var     sd
#> 1     prior     1    1 0.5000  NaN 0.08333 0.2887
#> 2 posterior    10   12 0.4545 0.45 0.01078 0.1038
```

Per calcolare la distribuzione a posteriori, ho qui usato le funzioni del pacchetto `bayesrules`. Ma per lo schema beta-binomiale è facile trovare i parametri della distribuzione a posteriori. Per esempio, nel caso dell'amica femminista, la distribuzione a posteriori è una Beta di parametri

$$\alpha_{post} = \alpha_{prior} + y = 5 + 9 = 14$$

e

$$\beta_{post} = \beta_{prior} + n - y = 11 + 20 - 9 = 22.$$

L'aggiornamento bayesiano indica che le tre amiche ottengono valori per la media (o la moda) a posteriori per π molto diversi. Dunque, anche dopo avere visto 20 film, le tre amiche non si trovano d'accordo su quale sia la proporzione di film che passano il test di Bechdel.

Questo non dovrebbe sorprenderci. L'amica ottimista aveva opinioni molto forti sul valore di π e i *pochi* nuovi dati che le sono stati forniti non sono riusciti a convincerla a cambiare idea: crede ancora che i valori $\pi > 0.5$ siano i più plausibili. Lo stesso si può dire, all'estremo opposto, dell'amica femminista: anche lei continua a credere che i valori $\pi <= .5$ siano i più plausibili. Infine, l'ignara non aveva nessuna opinione a priori su π e, anche dopo avere visto 20 film, continua a credere che il valore π più plausibile sia quello intermedio, nell'intorno di 0.5.

3.3 Dati diversi ma la stessa distribuzione a priori

Supponiamo ora che l'amica ottimista abbia tre amiche, Maria, Anna e Sara, tutte ottimiste come lei. L'ottimista chiede a Maria, Anna e Sara di fare loro stesse l'esperimento descritto in precedenza. Maria guarda 13 film; di questi 6 passano il test di Bechdel. Anna guarda 63 film; di questi 29 passano il test di Bechdel. Sara guarda 99 film; di questi 46 passano il test di Bechdel.

Supponiamo che Maria, Anna e Sara condividano la stessa credenza a priori su π: ovvero, Beta(14, 1). In tali circostanze e, alla luce dei dati osservati, cosa possiamo dire delle tre distribuzioni a posteriori?

```
p1 <- bayesrules:::plot_beta_binomial(
  alpha = 14, beta = 1, y = 6, n = 13
  ) +
  theme(legend.position = "none")
p2 <- bayesrules:::plot_beta_binomial(
  alpha = 14, beta = 1, y = 29, n = 63
  ) +
  theme(legend.position = "none")
p3 <- bayesrules:::plot_beta_binomial(
  alpha = 14, beta = 1, y = 46, n = 99
  ) +
  theme(legend.position = "none")
p1 + p2 + p3
```

Notiamo due cose. All'aumentare delle informazioni disponibili (ovvero,

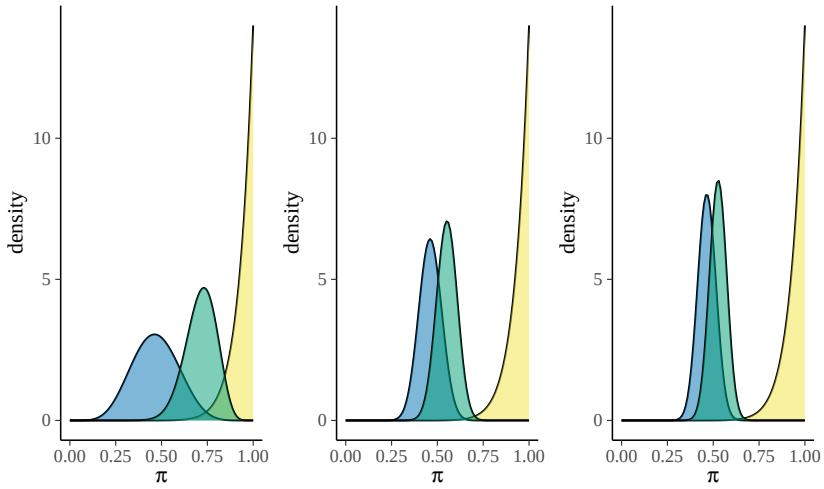


Figura 3.1: *Aggiornamento bayesiano per le credenze di Maria, Anna e Sara.*

all'aumentare dell'ampiezza del campione), la distribuzione a posteriori si allontana sempre di più dalla distribuzione a priori, e si avvicina sempre di più alla verosimiglianza. In secondo luogo, all'aumentare dell'ampiezza del campione la varianza della distribuzione a posteriori diminuisce sempre di più — ovvero, diminuisce l'incertezza su quelli che sono i valori π più plausibili.

3.4 Dati diversi e diverse distribuzioni a priori

Nella figura successiva esaminiamo le distribuzioni a posteriori che si ottengono incrociando tre diversi set di dati ($y = 6$, $n = 13$;, $y = 29$, $n = 63$; $y = 66$, $n = 99$) con tre diverse distribuzioni a priori [Beta(14, 1), Beta(5, 11), Beta(1, 1)].

```
p1 <- bayesrules:::plot_beta_binomial(
  alpha = 14, beta = 1, y = 6, n = 13
  ) +
  theme(legend.position = "none")
p2 <- bayesrules:::plot_beta_binomial(
```

```
  alpha = 14, beta = 1, y = 29, n = 63
  ) +
  theme(legend.position = "none")
p3 <- bayesrules:::plot_beta_binomial(
  alpha = 14, beta = 1, y = 46, n = 99
  ) +
  theme(legend.position = "none")
p4 <- bayesrules:::plot_beta_binomial(
  alpha = 5, beta = 11, y = 6, n = 13
  ) +
  theme(legend.position = "none")
p5 <- bayesrules:::plot_beta_binomial(
  alpha = 5, beta = 11, y = 29, n = 63
  ) +
  theme(legend.position = "none")
p6 <- bayesrules:::plot_beta_binomial(
  alpha = 5, beta = 11, y = 46, n = 99
  ) +
  theme(legend.position = "none")
p7 <- bayesrules:::plot_beta_binomial(
  alpha = 1, beta = 1, y = 6, n = 13
  ) +
  theme(legend.position = "none")
p8 <- bayesrules:::plot_beta_binomial(
  alpha = 1, beta = 1, y = 29, n = 63
  ) +
  theme(legend.position = "none")
p9 <- bayesrules:::plot_beta_binomial(
  alpha = 1, beta = 1, y = 46, n = 99
  ) +
  theme(legend.position = "none")
(p1 + p2 + p3) / (p4 + p5 + p6) / (p7 + p8 + p9)
```

La figura indica che, se il campione è grande, una distribuzione a priori debolmente informativa ha uno scarso effetto sulla distribuzione a posteriori. Invece, se il campione è piccolo, anche una distribuzione a priori debolmente informativa ha un grande effetto sulla distribuzione a posteriori.

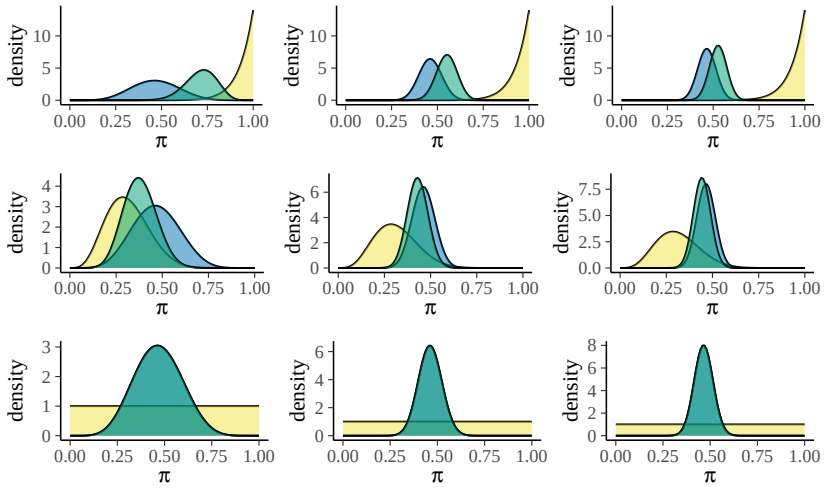


Figura 3.2: *Sulle colonne (a partire da sinistra) i dati utilizzati sono, rispettivamente, (y = 6, n = 13), (y = 29, n = 63) e (y = 66, n = 99). Sulle righe (a partire dall'alto), le distribuzioni a priori usate sono: Beta(14, 1), Beta(5, 11) e Beta(1, 1).*

3.5 Collegare le intuizioni alla teoria

Il compromesso che abbiamo osservato nell'esempio precedente, che combina la distribuzione a priori con le evidenze fornite dai dati, è molto vicino alle nostre intuizioni. Ma è anche il frutto di una necessità matematica. È infatti possibile riscrivere la (2.1) nel modo seguente

$$\begin{aligned} \mathbb{E}_{\text{post}}[\text{Beta}(\alpha + y, \beta + n - y)] &= \frac{\alpha + y}{\alpha + \beta + n} \\ &= \frac{a+b}{a+b+n} \cdot \frac{a}{a+b} + \frac{n}{a+b+n} \cdot \frac{y}{n}. \end{aligned} \tag{3.2}$$

Ciò indica che il valore atteso a posteriori è una media pesata fra il valore atteso a priori $\left(\frac{\alpha}{\alpha+\beta}\right)$ e la frequenze di successi osservata $\left(\frac{y}{n}\right)$. I pesi sono $\left(\frac{\alpha+\beta}{\alpha+\beta+n}\right)$ e $\left(\frac{n}{\alpha+\beta+n}\right)$. Quindi, quando n è grande rispetto ad $\alpha + \beta$, conta molto quanto abbiamo osservato e conta poco la credenza a priori. Viceversa, quando n è piccolo rispetto a $\alpha + \beta$, le osservazioni contano poco rispetto alla credenza a priori.

Queste osservazioni ci fanno capire come scegliere i parametri α e β: se vogliamo assumere una totale ignoranza rispetto al fenomeno in esame, la scelta coerente è $\alpha = \beta = 1$ (ogni valore di θ è ugualmente probabile); se invece abbiamo delle credenze a priori, allora possiamo scegliere α così che sia uguale al valore atteso a priori, mentre $\alpha+\beta$ esprime l'importanza che diamo all'informazione a priori: maggiore è il valore di $\alpha + \beta$, tanti più dati serviranno per allontanare la distribuzione a posteriori dalla distribuzione a priori. Se n è grande, infine, la distribuzione a posteriori sarà scarsamente influenzata dalla distribuzione a priori, a meno di scelte estreme.

Commenti e considerazioni finali

La conclusione che possiamo trarre dall'esempio di Johnson et al. (2022) è molto chiara: l'aggiornamento bayesiano può essere paragonato ai processi di ragionamento del senso comune. Quando le nuove evidenze (i dati) sono deboli, non c'è ragione di cambiare idea (le nostre credenze "a posteriori" sono molto simili a ciò che pensavamo prima di avere osservato i dati). Quando le nuove evidenze sono irrefutabili, invece, è necessario modificare le nostre credenze sulla base di ciò che ci dicono i dati, quali che siano le nostre credenze pregresse — non farlo significherebbe vivere in un mondo di fantasia e avere scarse possibilità di sopravvivere nel mondo empirico. L'aggiornamento bayesiano esprime in maniera quantitativa e precisa ciò che ci dicono le nostre intuizioni.

Incredibilmente, però, l'approccio frequentista nega questa logica. I test frequentisti non tengono conto delle conoscenze pregresse. Dunque, se un test frequentista, calcolato su un piccolo campione (ovvero, quando i dati sono molto deboli), suggerisce che dovremmo farci un'opinione di un certo tipo sul fenomeno in esame, l'indicazione è di prendere seriamente il risultato del test *quali siano le evidenze precedenti* — le quali, possibilmente, mostrano che il risultato del test non ha alcun senso. È sorprendente che un tale modo di pensare possa essere preso sul serio nella comunità scientifica, ma vi sono alcuni ricercatori che continuano a seguire questo modo di (s)ragionare. Dato che in questo Capitolo paliamo di fumetti, concluderei dicendo che la presente discussione è catturata

nella maniera più chiara possibile in questa famosa striscia[1].

[1] https://xkcd.com/1132/

4

Approssimazione della distribuzione a posteriori

In generale, in un problema bayesiano i dati y provengono da una densità $p(y \mid \theta)$ e al parametro θ viene assegnata una densità a priori $p(\theta)$. Dopo avere osservato i dati $Y = y$, la funzione di verosimiglianza è uguale a $\mathcal{L}(\theta) = p(y \mid \theta)$ e la densità a posteriori diventa

$$p(\theta \mid y) = \frac{p(y \mid \theta)p(\theta)}{\int p(y \mid \theta)p(\theta) \, \mathrm{d}\theta}.$$

Si noti che, quando usiamo il teorema di Bayes per calcolare la distribuzione a posteriori del parametro di un modello statistico, al denominatore troviamo un integrale. Se vogliamo trovare la distribuzione a posteriori con metodi analitici è necessario usare distribuzioni a priori coniugate per la verosimiglianza, come nello schema beta-binomiale. Per quanto "semplice" in termini formali, la scelta di una famiglia di distribuzione a priori tale che la distribuzione a posteriori sia della stessa famiglia limita di molto le possibili scelte del ricercatore. Inoltre, non è sempre sensato, dal punto di vista teorico, utilizzare distribuzioni a priori coniugate per la verosimiglianza per i parametri di interesse. Se non vengono usate distribuzioni a priori coniutate per la verosimiglianza, la determinazione della distribuzione a posteriori richiede il calcolo di un integrale che, nella maggior parte dei casi, non si può risolvere analiticamente. In altre parole, è possibile ottenere analiticamenre la distribuzione a posteriori solo per alcune specifiche combinazioni di distribuzioni a priori e verosimiglianza, il che limita considerevolmente la flessibilità della modellizzazione. Inoltre, i sommari della distribuzione a posteriori sono espressi come rapporto di integrali. Ad esempio, la media a posteriori di θ è data da

$$\mathbb{E}(\theta \mid y) = \frac{\int \theta p(y \mid \theta)p(\theta) \, \mathrm{d}\theta}{\int p(y \mid \theta)p(\theta) \, \mathrm{d}\theta}.$$

Il calcolo del valore atteso a posteriori richiede dunque la valutazione di due integrali, ciascuno dei quali non esprimibile in forma chiusa.

Per questa ragione, la strada principale che viene seguita nella modellistica bayesiana è quella che porta a determinare la distribuzione a posteriori non per via analitica, ma bensì mediante metodi numerici. La simulazione fornisce dunque la strategia generale del calcolo bayesiano. A questo fine vengono principalmente usati i metodi di campionamento Monte Carlo basati su Catena di Markov (MCMC). Tali metodi costituiscono una potente e praticabile alternativa per la costruzione della distribuzione a posteriori per modelli complessi e consentono di decidere quali distribuzioni a priori e quali distribuzioni di verosimiglianza usare sulla base di considerazioni teoriche soltanto, senza dovere preoccuparsi di altri vincoli.

Dato che è basata su metodi computazionalmente intensivi, la stima numerica della funzione a posteriori può essere svolta soltanto mediante software. In anni recenti i metodi Bayesiani di analisi dei dati sono diventati sempre più popolari proprio perché la potenza di calcolo necessaria per svolgere tali calcoli è ora alla portata di tutti. Questo non era vero solo pochi decenni fa.

In generale, possiamo distinguere cinque metodi per calcolare le proprietà di una distribuzione (Lunn et al., 2013).

1. **Metodi analitici esatti:** per esempio, nel caso delle famiglie coniugate, quando disponiamo di una forma analitica della distribuzione a posteriori.
2. **Metodi numerici esatti:** dove, sebbene non sia disponibile alcuna formula algebrica in forma chiusa, le proprietà della distribuzione a posteriori possono essere calcolate con una precisione arbitraria – vedremo un esempio di come questo possa essere fatto nella seguente discussione sui metodi basati su griglia.
3. **Metodi analitici approssimati:** per esempio, nel caso di approssimazioni normali alle distribuzioni di variabili casuali – in questo Capitolo verrà discussa l'approssimazione quadratica della distribuzione a posteriori che è un esempio di questo approccio.
4. **Sperimentazione fisica:** ad esempio, quando un esperimento viene fisicamente ripetuto molte volte per determinare la

proporzione empirica dei "successi".

5. **Simulazione al computer:** utilizzando appropriate funzioni di numeri casuali, viene generato un ampio campione di casi della variabile casuale per poi stimare empiricamente la proprietà di interesse in base al campione così otttenuto. Questa tecnica è conosciuta come metodo di Monte Carlo ed è il metodo che verrà utilizzato nella seconda parte di questa dispensa.

In questo Capitolo ci focalizzeremo su tre tecniche numeriche per il calcolo della distribuzione a posteriori:

1. il metodo basato su griglia;
2. il metodo dell'approssimazione quadratica,
3. il metodo basato su simulazione di Monte Carlo basato su Catena di Markov (*Markov Chain Monte Carlo*, MCMC).

4.1 Metodo basato su griglia

Il metodo basato su griglia (*grid-based*) è un metodo numerico esatto basato su una griglia di punti uniformemente spaziati. Anche se la maggior parte dei parametri è continua (ovvero, in linea di principio ciascun parametro può assumere un numero infinito di valori), possiamo ottenere un'eccellente approssimazione della distribuzione a posteriori considerando solo una griglia finita di valori dei parametri. In un tale metodo, la densità di probabilità a posteriori può dunque essere approssimata tramite le densità di probabilità calcolate in ciascuna cella della griglia.

Il metodo basato su griglia si sviluppa in quattro fasi:

- fissare una griglia discreta di possibili valori θ;
- valutare la distribuzione a priori $p(\theta)$ e la funzione di verosimiglianza $p(y \mid \theta)$ in corrispondenza di ciascun valore θ della griglia;
- ottenere un'approssimazione discreta della densità a posteriori:
  - per ciascun valore θ della griglia, calcolare il prodotto $p(\theta)p(y \mid \theta)$;
  - normalizzare i prodotti così ottenuti in modo tale che la loro somma sia 1;
- selezionare N valori casuali della griglia in modo tale da ottenere un campione casuale delle densità a posteriori normalizzate.

Possiamo migliorare l'approssimazione aumentando il numero di punti della griglia. Infatti utilizzando un numero infinito di punti si otterrebbe la descrizione esatta della distribuzione a posteriori, dovendo però pagare il costo dell'utilizzo di infinite risorse di calcolo. Il limite maggiore dell'approccio basato su griglia è che, al crescere della dimensionalità N dello spazio dei parametri, i punti della griglia necessari per avere una buona stima crescerebbero esponenzialmente con N, rendendo questo metodo inattuabile.

4.1.1 Modello Beta-Binomiale

Per fare un esempio, consideriamo lo schema beta-binomiale di cui conosciamo la soluzione esatta. Utilizziamo nuovamente i dati di Zetsche et al. (2019): 23 "successi" in 30 prove Bernoulliane indipendenti.[1] Imponiamo alla distribuzione a priori su θ (probabilità di successo in una singola prova, laddove per "successo" si intende una aspettativa distorta negativamente dell'umore futuro) una Beta$(2, 10)$ per descrivere la nostra incertezza sul parametro prima di avere osservato i dati. Dunque, il modello diventa:

$$\begin{aligned} Y \mid \theta &\sim \text{Bin}(n = 30, \theta), \\ \theta &\sim \text{Beta}(2, 10). \end{aligned}$$

In queste circostanze, l'aggiornamento bayesiano produce una distribuzione a posteriori Beta di parametri 25 ($y + \alpha = 23 + 2$) e 17 ($n - y + \beta$ = 30 - 23 + 10):

$$\theta \mid (y = 23) \sim \text{Beta}(25, 17).$$

Per approssimare la distribuzione a posteriori, fissiamo una griglia di $n = 11$ valori equispaziati: $\theta \in \{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\}$:

```
grid_data <- tibble(
  theta_grid = seq(from = 0, to = 1, length.out = 11)
)
grid_data
```

[1] Nel presente esempio useremo lo stesso codice R utilizzato da Johnson et al. (2022).

```
#> # A tibble: 11 x 1
#>   theta_grid
#>        <dbl>
#> 1        0
#> 2        0.1
#> 3        0.2
#> 4        0.3
#> 5        0.4
#> 6        0.5
#> 7        0.6
#> 8        0.7
#> # ... with 3 more rows
```

In corrispondenza di ciascun valore della griglia, valutiamo la distribuzione a priori Beta$(2, 10)$ e la verosimiglianza Bin$(y = 23, n = 30)$.

```
grid_data <- grid_data %>%
  mutate(
    prior = dbeta(theta_grid, 2, 10),
    likelihood = dbinom(23, 30, theta_grid)
  )
```

In ciascuna cella della griglia calcoliamo poi il prodotto della verosimiglianza e della distribuzione a priori. Troviamo così un'approssimazione discreta e non normalizzata della distribuzione a posteriori (`unnormalized`). Normalizziamo questa approssimazione dividendo ciascun valore `unnormalized` per la somma di tutti i valori del vettore:

```
grid_data <- grid_data %>%
  mutate(
    unnormalized = likelihood * prior,
    posterior = unnormalized / sum(unnormalized)
  )
```

Verifichiamo:

```
grid_data %>%
  summarize(
    sum(unnormalized),
    sum(posterior)
  )
#> # A tibble: 1 x 2
#>   `sum(unnormalized)` `sum(posterior)`
#>                 <dbl>            <dbl>
#> 1            0.000869                1
```

Abbiamo dunque ottenuto la seguente distribuzione a posteriori discretizzata $p(\theta \mid y)$:

```
round(grid_data, 2)
#> # A tibble: 11 x 5
#>   theta_grid prior likelihood unnormalized posterior
#>        <dbl> <dbl>      <dbl>        <dbl>     <dbl>
#> 1        0    0          0               0      0
#> 2        0.1  4.26       0               0      0
#> 3        0.2  2.95       0               0      0
#> 4        0.3  1.33       0               0      0
#> 5        0.4  0.44       0               0      0.02
#> 6        0.5  0.11       0               0      0.23
#> 7        0.6  0.02       0.03            0      0.52
#> 8        0.7  0          0.12            0      0.21
#> # ... with 3 more rows
```

La figura 4.1 mostra un grafico della distribuzione a posteriori discretizzata così ottenuta:

```
grid_data %>%
  ggplot(
    aes(x = theta_grid, y = posterior)
  ) +
  geom_point() +
  geom_segment(
    aes(
      x = theta_grid,
```

```
    xend = theta_grid,
    y = 0,
    yend = posterior)
)
```

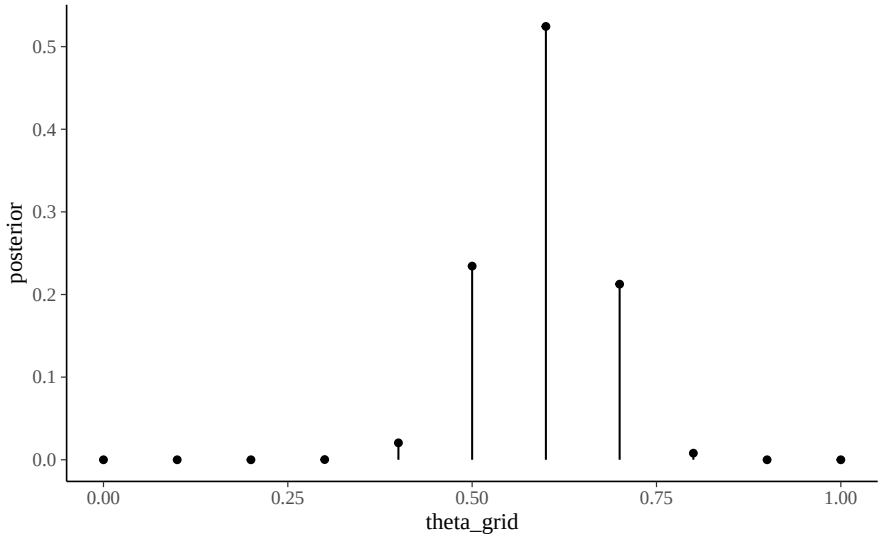


Figura 4.1: *Distribuzione a posteriori discretizzata ottenuta con il metodo grid-based per $y = 23$ successi in 30 prove Bernoulliane, con distribuzione a priori $Beta(2, 10)$. È stata utilizzata una griglia di solo $n = 11$ punti.*

L'ultimo passo della simulazione è il campionamento dalla distribuzione a posteriori discretizzata:

```
set.seed(84735)
post_sample <- sample_n(
  grid_data,
  size = 1e5,
  weight = posterior,
  replace = TRUE
)
```

La figura 4.2 mostra che, con una griglia così sparsa abbiamo ottenuto una versione approssimata della vera distribuzione a posteriori

(all'istogramma è stata sovrapposta l'esatta distribuzione a posteriori Beta$(25, 17)$).

```
ggplot(post_sample, aes(x = theta_grid)) +
  geom_histogram(aes(y = ..density..), color = "white") +
  stat_function(fun = dbeta, args = list(25, 17)) +
  lims(x = c(0, 1))
```

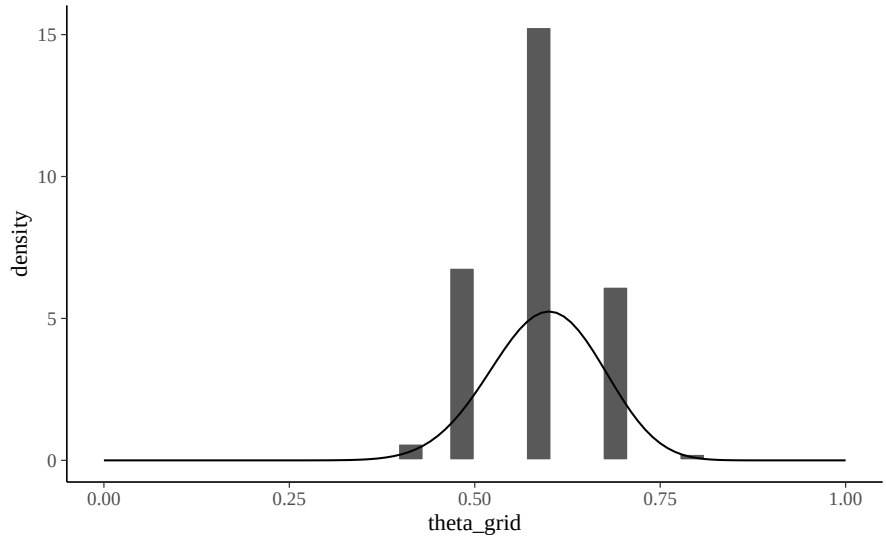


Figura 4.2: *Campionamento dalla distribuzione a posteriori discretizzata ottenuta con il metodo grid-based per $y = 23$ successi in 30 prove Bernoulliane, con distribuzione a priori Beta$(2, 10)$. È stata utilizzata una griglia di solo $n = 11$ punti.*

Possiamo ottenere un risultato migliore con una griglia più fine, come indicato nella figura 4.3:

```
grid_data <- tibble(
  theta_grid = seq(from = 0, to = 1, length.out = 100)
)
grid_data <- grid_data %>%
  mutate(
    prior = dbeta(theta_grid, 2, 10),
    likelihood = dbinom(23, 30, theta_grid)
```

```
  )
grid_data <- grid_data %>%
  mutate(
    unnormalized = likelihood * prior,
    posterior = unnormalized / sum(unnormalized)
  )
grid_data %>%
  ggplot(
    aes(x = theta_grid, y = posterior)
  ) +
  geom_point() +
  geom_segment(
    aes(
      x = theta_grid,
      xend = theta_grid,
      y = 0,
      yend = posterior
    )
  )
```

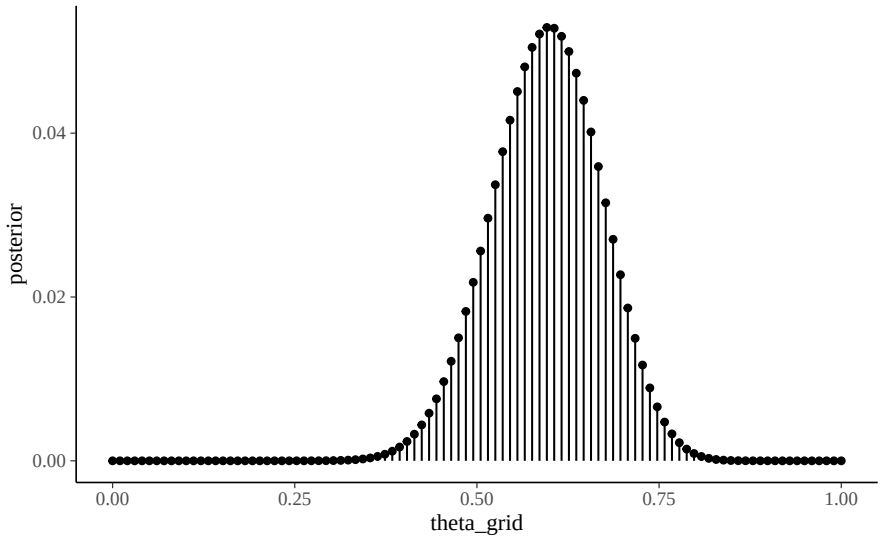


Figura 4.3: *Distribuzione a posteriori discretizzata ottenuta con il metodo grid-based per $y = 23$ successi in 30 prove Bernoulliane, con distribuzione a priori Beta$(2, 10)$. È stata utilizzata una griglia di $n = 100$ punti.*

Campioniamo ora 10000 punti:

```
# Set the seed
set.seed(84735)
post_sample <- sample_n(
  grid_data,
  size = 1e4,
  weight = posterior,
  replace = TRUE
)
```

Con il campionamento dalla distribuzione a posteriori discretizzata costruita mediante una griglia più densa ($n = 100$) otteniamo un risultato soddisfacente (figura 4.4): ora la distribuzione dei valori prodotti dalla simulazione approssima molto bene la corretta distribuzione a posteriori $p(\theta \mid y) = \text{Beta}(25, 17)$.

```
post_sample %>%
  ggplot(aes(x = theta_grid)) +
  geom_histogram(
    aes(y = ..density..),
    color = "white",
    bins=50
  ) +
  stat_function(fun = dbeta, args = list(25, 17)) +
  lims(x = c(0, 1))
```

In conclusione, il metodo basato su griglia è molto intuitivo e non richiede particolari competenze di programmazione per essere implementato. Inoltre, fornisce un risultato che, per tutti gli scopi pratici, può essere considerato come un campione casuale estratto da $p(\theta \mid y)$. Tuttavia, anche se tale metodo fornisce risultati accuratissimi, esso ha un uso limitato. A causa della *maledizione della dimensionalità*[2], tale metodo può

[2]Per capire cosa sia la maledizione della dimensionalità, supponiamo di utilizzare una griglia di 100 punti equispaziati. Nel caso di un solo parametro, è necessario calcolare 100 valori. Per due parametri devono essere calcolari 100^2 valori. Ma già per 10 parametri è necessario calcolare 10^{10} valori – è facile capire che una tale quantità di calcoli è troppo grande anche per un computer molto potente. Per modelli che richiedono la stima di un numero non piccolo di parametri è dunque necessario procedere in un altro modo.

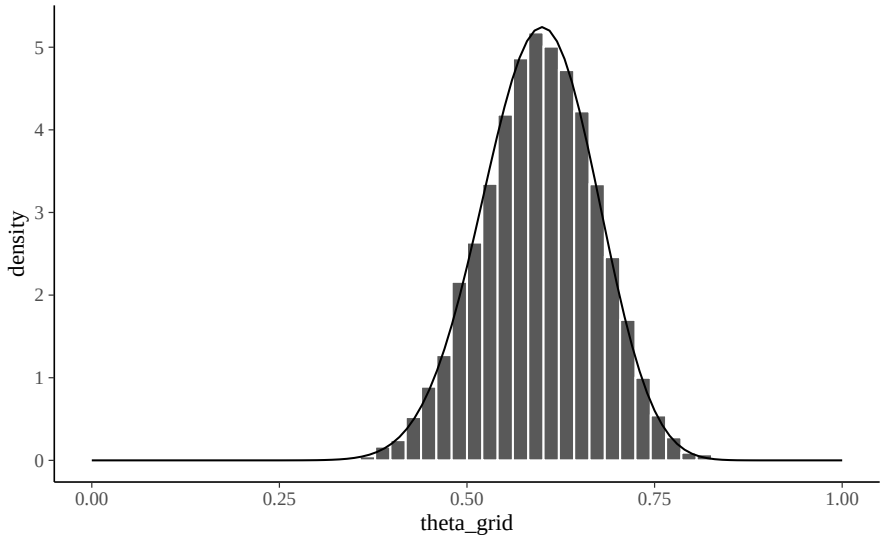


Figura 4.4: *Campionamento dalla distribuzione a posteriori discretizzata ottenuta con il metodo grid-based per $y = 23$ successi in 30 prove Bernoulliane, con distribuzione a priori Beta$(2, 10)$. È stata utilizzata una griglia di $n = 100$ punti. All'istogramma è stata sovrapposta la corretta distribuzione a posteriori, ovvero la densità Beta$(25, 17)$.*

solo essere solo nel caso di semplici modelli statistici, con non più di due parametri. Nella pratica concreta tale metodo viene dunque sostituito da altre tecniche più efficienti in quanto, anche nei più comuni modelli utilizzati in psicologia, vengono solitamente stimati centinaia se non migliaia di parametri.

4.2 Approssimazione quadratica

L'approssimazione quadratica è un metodo analitico approssimato che può essere usato per superare il problema della "maledizione della dimensionalità". La motivazione di tale metodo è la seguente. Sappiamo che, in generale, la regione della distribuzione a posteriori che si trova in prossimità del suo massimo può essere ben approssimata dalla forma di

una distribuzione gaussiana.[3]

L'approssimazione quadratica si pone due obiettivi.

1. Trovare la moda della distribuzione a posteriori. Ci sono varie procedure di ottimizzazione, implementate in R, in grado di trovare il massimo di una distribuzione.
2. Stimare la curvatura della distribuzione in prossimità della moda. Una stima della curvatura è sufficiente per trovare un'approssimazione quadratica dell'intera distribuzione. In alcuni casi, questi calcoli possono essere fatti seguendo una procedura analitica, ma solitamente vengono usate delle tecniche numeriche.

Per fare un esempio concreto, consideriamo nuovamente i dati di Zetsche et al. (2019) (ovvero, 23 "successi" in 30 prove Bernoulliane). Supponiamo di usare una Beta$(2, 10)$ quale distribuzione a priori per il parametro sconosciuto θ (probabilità che l'aspettativa dell'umore futuro sia distorta negativamente). Una descrizione della distribuzione a posteriori ottenuta mediante l'approssimazione quadratica si ottiene utilizzando la funzione `quap()` del pacchetto `rethinking`:[4] Dopo avere specificato il modello utilizzando la sintassi appropriata

```
suppressPackageStartupMessages(library("rethinking"))

mod <- quap(
  alist(
  N ~ dbinom(N + P, p),
  p ~ dbeta(2, 10)
  ),
  data = list(N = 23, P = 7)
)
```

[3] Descrivere la distribuzione a posteriori mediante la distribuzione gaussiana significa utilizzare un'approssimazione che viene, appunto, chiamata "quadratica". Tale approssimazione si dice quadratica perché il logaritmo di una distribuzione gaussiana forma una parabola e la parabola è una funzione quadratica – dunque, mediante questa approssimazione descriviamo il logaritmo della distribuzione a posteriori mediante una parabola.

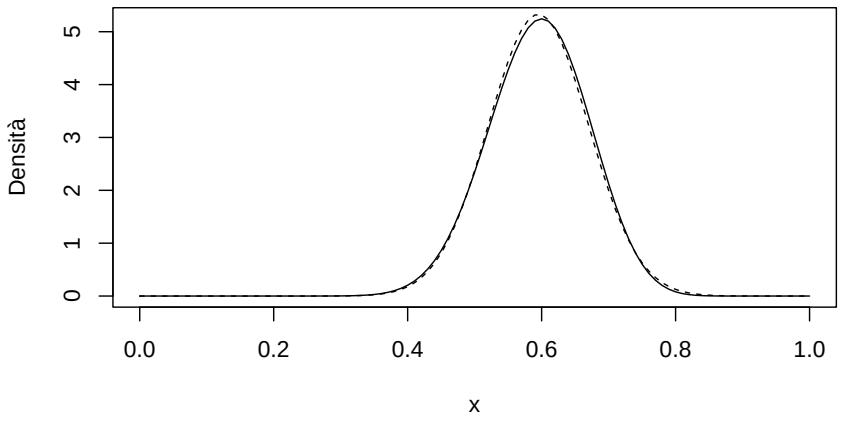
[4] Il pacchetto `rethinking` è stato creato da McElreath (2020) per accompagnare il suo testo *Statistical Rethinking*[2]. Per l'installazione si veda https://github.com/rmcelreath/rethinking.

un sommario dell'approssimazione quadratica si ottiene con la funzione `precis()`:

```
precis(mod, prob = 0.95)
#>   mean      sd   2.5%  97.5%
#> p  0.6 0.07746 0.4482 0.7518
```

La figura seguente fornisce un confronto tra la corretta distribuzione a posteriori (linea continua) e l'approssimazione quadratica (linea trateggiata).

```
N <- 23
P <- 7
a <- N + 2
b <- P + 10
curve(dbeta(x, a, b), from=0, to=1, ylab="Densità")
# approssimazione quadratica
curve(
  dnorm(x, a/(a+b), sqrt((a*b)/((a+b)^2*(a+b+1)))),
  lty = 2,
  add = TRUE
)
```



Il grafico precedente mostra che, nel caso dell'esempio, l'approssimazione quadratica fornisce un risultato soddisfacente. Tale risultato è quasi identico a quello che può essere trovato con il metodo *grid-based*, con il vantaggio aggiuntivo che abbiamo accesso ad una serie di funzioni R in grado di svolgere i calcoli per noi. In realtà, però, l'approssimazione quadratica è poco usata perché, per problemi complessi, è più conveniente fare ricorso ai metodi MCMC che verranno descritti nel Paragrafo successivo.

4.3 Metodo Monte Carlo

I metodi più ampiamente adottati nell'analisi bayesiana per la costruzione della distribuzione a posteriori per modelli complessi sono i metodi di campionamento MCMC. Tali metodi consentono al ricercatore di decidere quali distribuzioni a priori e quali distribuzioni di verosimiglianza usare sulla base di considerazioni teoriche soltanto, senza doversi preoccupare di altri vincoli. Dato che è basata su metodi computazionalmente intensivi, la stima numerica MCMC della funzione a posteriori può essere svolta soltanto mediante software. In anni recenti i metodi Bayesiani di analisi dei dati sono diventati sempre più popolari proprio perché la potenza di calcolo necessaria per svolgere tali calcoli è alla portata di

tutti. Questo non era vero solo pochi decenni fa.

4.3.1 Integrazione di Monte Carlo

Il termine Monte Carlo si riferisce al fatto che la computazione fa ricorso ad un ripetuto campionamento casuale attraverso la generazione di sequenze di numeri casuali. Una delle sue applicazioni più potenti è il calcolo degli integrali mediante simulazione numerica. Data una successione di realizzazioni indipendenti $y^{(1)}, y^{(2)}, \ldots, y^{(T)}$ da una distribuzione $p(y)$ con media μ abbiamo che

$$\mathbb{E}(Y) = \int yp(y)\ \mathrm{d}y \approx \frac{1}{T}\sum_{i=1}^{T} y^{(t)}.$$

In altre parole, l'aspettazione teorica di Y può essere approssimata dalla media campionaria di un insieme di realizzazioni indipendenti ricavate da $p(y)$. Per la Legge Forte dei Grandi Numeri, l'approssimazione diventa arbitrariamente esatta per $T \to \infty$. L'integrazione Monte Carlo può essere utilizzata anche per la valutazione di integrali più complessi.

Quello che è stato detto sopra non è altro che un modo sofisticato per dire che, se vogliamo calcolare un'approssimazione del valore atteso di una variabile casuale, non dobbiamo fare altro che la media aritmetica di un grande numero di realizzazioni indipendenti della variabile casuale. Come è facile intuire, l'approssimazione migliora al crescere del numero di dati che abbiamo a disposizione.

Un'altra importante funzione di Y è la funzione indicatore, $I(l < Y < u)$, che assume valore 1 se Y giace nell'intervallo (l, u) e 0 altrimenti. Il valore di aspettazione di $I(l < X < u)$ rispetto a $p(x)$ dà la probabilità che Y rientri nell'intervallo specificato, $Pr(l < T < u)$, e può essere approssimato usando l'integrazione Monte Carlo, ovvero prendendo la media campionaria del valore della funzione indicatore per ogni realizzazione $y^{(t)}$. È semplice vedere come

$$Pr(l < Y < u) \approx \frac{\text{numero di realizzazioni } y^{(t)} \in (l, u)}{T}.$$

Presentiamo qui l'integrazione di Monte Carlo perché può essere usata per approssimare la distribuzione a posteriori richiesta da un'analisi Bayesiana: una stima di $p(\theta \mid y)$ viene infatti ottenuta mediante

un grande numero di campioni casuali estratti dalla distribuzionea a posteriori.

4.3.2 Un esempio concreto

Per introdurre i metodi MCMC consideriamo nuovamente i dati di Zetsche et al. (2019) (23 "successi" in 30 prove Bernoulliane) che possono essere analizzati mediante lo schema beta-binomiale. In questo caso, la distribuzione a posteriori può essere ottenuta analiticamente ed è uguale ad una Beta$(25, 17)$. Se vogliamo trovare il valore della media a posteriori di θ, il risultato esatto è

$$\bar{\theta}_{post} = \frac{\alpha}{\alpha + \beta} = \frac{25}{25 + 17} \approx 0.5952.$$

In alternativa, sapendo che la distribuzione a posteriori di θ è Beta$(25, 17)$, possiamo estrarre un campione casuale di osservazioni da tale distribuzione e calcolare la media:

```
set.seed(7543897)
print(mean(rbeta(1e2, shape1 = 25, shape2 = 17)), 6)
#> [1] 0.587548
```

È ovvio che l'approssimazione migliora all'aumentare del numero di osservazioni estratte dalla distribuzione a posteriori (legge dei grandi numeri):

```
print(mean(rbeta(1e3, shape1 = 25, shape2 = 17)), 6)
#> [1] 0.597659
print(mean(rbeta(1e4, shape1 = 25, shape2 = 17)), 6)
#> [1] 0.595723
print(mean(rbeta(1e5, shape1 = 25, shape2 = 17)), 6)
#> [1] 0.595271
```

Quando il numero di osservazioni (possiamo anche chiamarle "campioni") tratte dalla distribuzione a posteriori è molto grande, la distribuzione di tali campioni converge alla densità della popolazione (si veda l'Appendice 4.3.1).[5]

[5]Si noti che il numero dei campioni di simulazione è controllato dal ricercatore;

Le statistiche descrittive (media, moda, varianza, eccetera) dei campioni estratti dalla distribuzione a posteriori convergeranno ai corrispondenti valori della distribuzione a posteriori. La figura 4.5 mostra come, all'aumentare del numero di repliche, la media, la mediana, la deviazione standard e l'asimmetria convergono ai veri valori della distribuzione a posteriori (linee rosse tratteggiate).

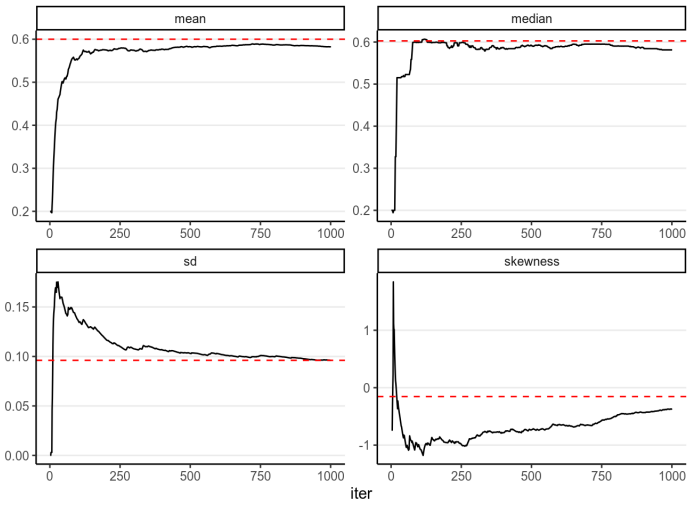


Figura 4.5: *Convergenza delle simulazioni Monte Carlo.*

4.3.3 Metodi MC basati su Catena di Markov

Nel Paragrafo 4.3 la simulazione Monte Carlo produce il risultato desiderato perché

- sappiamo che la distribuzione a posteriori è una Beta$(25, 17)$,
- è possibile usare le funzioni R per estrarre campioni casuali da tale distribuzione.

Tuttavia, capita raramente di usare una distribuzione a priori coniugata alla verosimiglianza. Quindi, in generale, le due condizioni descritte sopra non si applicano. Ad esempio, nel caso di una verosimiglianza binomiale e una distribuzione a priori Normale, la distribuzione a posteriori di θ è

tale valore è totalmente diverso dalla dimensione del campione che è fissa ed è una proprietà dei dati.

$$p(\theta \mid y) = \frac{\mathrm{e}^{-(\theta-1/2)^2}\theta^y(1-\theta)^{n-y}}{\int_0^1 \mathrm{e}^{-(t-1/2)^2}t^y(1-t)^{n-y}dt}.$$

Una tale distribuzione non è implementata in R e dunque non possiamo campionare da $p(\theta \mid y)$.

Il vantaggio degli algoritmi MCMC è che essi consentono il campionamento da una distribuzione a posteriori *senza che sia necessario conoscere la rappresentazione analitica di una tale distribuzione.*

I metodi MCMC consentono di costruire sequenze di punti (detti catene di Markov) nello spazio dei parametri le cui densità sono proporzionali alla distribuzione a posteriori — in altre parole, dopo aver simulato un grande numero di passi della catena si possono usare i valori così generati come se fossero un campione casuale della distribuzione a posteriori. Le tecniche MCMC sono attualmente il metodo computazionale maggiormente utilizzato per risolvere i problemi di inferenza bayesiana. Un'introduzione alle catene di Markov è fornita nell'Appendice **??**.

4.3.4 Campionamento mediante algoritmi MCMC

Un modo generale per ottenere una catena di Markov la cui distribuzione equivale alla distribuzione a posteriori $p(\theta \mid y)$ è quello di usare l'algoritmo di Metropolis. L'algoritmo di Metropolis è il primo algoritmo MCMC che è stato proposto, ed è applicabile ad una grande varietà di problemi inferenziali di tipo bayesiano. Questo algoritmo è stato in seguito sviluppato allo scopo di renderlo via via più efficiente.

4.3.5 Una passeggiata casuale sui numeri naturali

Per introdurre l'algoritmo di di Metropolis in una forma intuitiva considereremo ora il campionamento da una distribuzione discreta.[6] Supponiamo di definire una distribuzione di probabilità discreta sugli interi $1, \dots, K$. Scriviamo in R la funzione `pd()` che assegna ai valori $1, \dots, 8$ delle probabilità proporzionali a 5, 10, 4, 4, 20, 20, 12 e 5.

[6] Seguiamo qui la trattazione di Albert and Hu (2019). Per una presentazione intuitiva dell'algoritmo di Metropolis, si vedano anche Kruschke (2014); McElreath (2020).

```
pd <- function(x){
  values <- c(5, 10, 4, 4, 20, 20, 12, 5)
  ifelse(
    x %in% 1:length(values),
    values[x] / sum(values),
    0
  )
}
prob_dist <- tibble(
  x = 1:8,
  prob = pd(1:8)
)
```

La figura 4.6 illustra la distribuzione di probabilità che è stata generata.

```
x <- 1:8
prob_dist %>%
  ggplot(aes(x = x, y = prob)) +
  geom_bar(stat = "identity", width = 0.06) +
  scale_x_continuous("x", labels = as.character(x), breaks = x) +
  labs(
    y = "Probabilità",
    x = "X"
  )
```

L'algoritmo di Metropolis corrisponde alla seguente passeggiata casuale.

1. L'algoritmo inizia con un valore iniziale qualsiasi da 1 a $K = 8$ della variabile casuale.
2. Per simulare il valore successivo della sequenza, lanciamo una moneta equilibrata. Se esce testa, consideriamo come valore candidato il valore immediatamente precedente al valore corrente nella sequenza $1, \ldots, 8$; se esce croce, il valore candidato sarà il valore immediatamente successivo al valore corrente nella sequenza.
3. Calcoliamo il rapporto tra la probabilità del valore candidato e la probabilità del valore corrente:

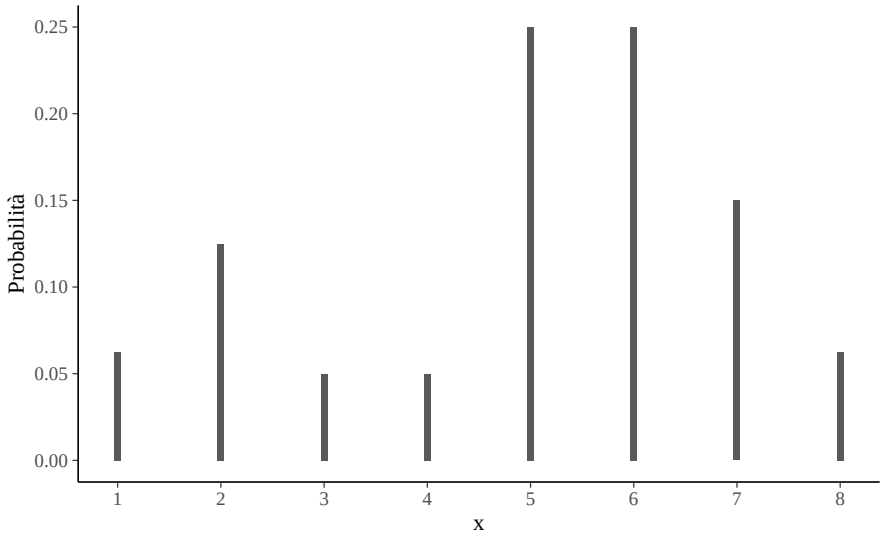


Figura 4.6: *Distribuzione di massa di probabilità per una variabile casuale avente valori 1, 2, ..., 8.*

$$R = \frac{pd(\text{valore candidato})}{pd(\text{valore corrente})}.$$

4. Estraiamo un numero a caso $\in [0, 1]$. Se tale valore è minore di R accettiamo il valore candidato come valore successivo della catena markoviana; altrimenti il valore successivo della catena rimane il valore corrente.

I passi da 1 a 4 definiscono una catena di Markov irriducibile e aperiodica sui valori di stato $\{1, 2, ..., 8\}$, dove il passo 1 fornisce il valore iniziale della catena e i passi da 2 a 4 definiscono la matrice di transizione P. Un modo di campionare da una distribuzione di massa di probabilità `pd` consiste nell'iniziare da una posizione qualsiasi e eseguire una passeggiata casuale costituita da un grande numero di passi, ripetendo le fasi 2, 3 e 4 dell'algoritmo di Metropolis. Dopo un grande numero di passi, la distribuzione dei valori della catena markoviana approssimerà la distribuzione di probabilità `pd`.

La funzione `random_walk()` implementa l'algoritmo di Metropolis. Tale funzione richiede in input la distribuzione di probabilità `pd`, la posizione di partenza `start` e il numero di passi dell'algoritmo `num_steps`.

```
random_walk <- function(pd, start, num_steps){
  y <- rep(0, num_steps)
  current <- start
  for (j in 1:num_steps){
    candidate <- current + sample(c(-1, 1), 1)
    prob <- pd(candidate) / pd(current)
    if (runif(1) < prob)
      current <- candidate
    y[j] <- current
  }
  return(y)
}
```

Di seguito, implementiamo l'algoritmo di Metropolis utilizzando, quale valore iniziale, $X = 4$. Ripetiamo la simulazione 10,000 volte.

```
out <- random_walk(pd, 4, 1e4)

S <- tibble(out) %>%
  group_by(out) %>%
  summarize(
    N = n(),
    Prob = N / 10000
  )

prob_dist2 <- rbind(
  prob_dist,
  tibble(
    x = S$out,
    prob = S$Prob
  )
)
prob_dist2$Type <- rep(
  c("Prob. corrette", "Prob. simulate"),
  each = 8
)
```

```
x <- 1:8
prob_dist2 %>%
  ggplot(aes(x = x, y = prob, fill = Type)) +
  geom_bar(
    stat = "identity",
    width = 0.1,
    position = position_dodge(0.3)
  ) +
  scale_x_continuous(
    "x",
    labels = as.character(x),
    breaks = x
  ) +
  scale_fill_manual(values = c("black", "gray80")) +
  theme(legend.title = element_blank()) +
  labs(
    y = "Probabilità",
    x = "X"
  )
```

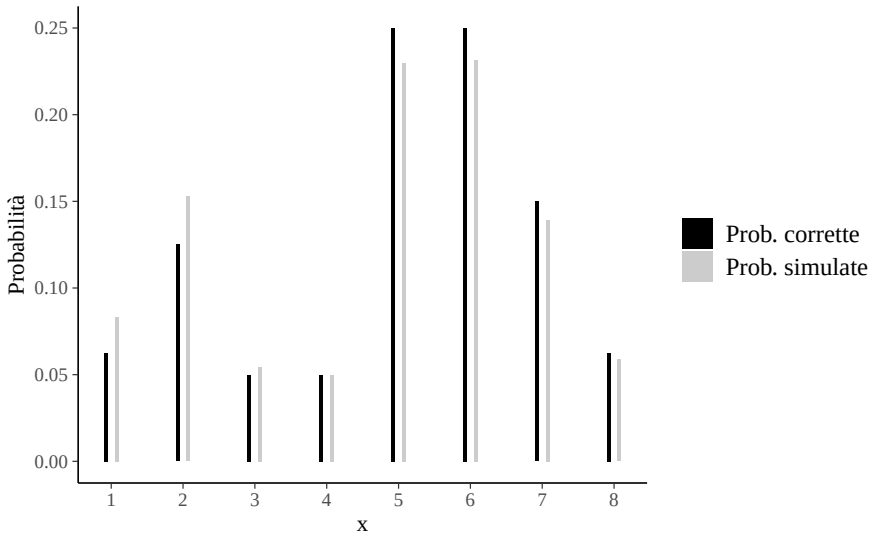


Figura 4.7: *L'istogramma confronta i valori prodotti dall'algoritmo di Metropolis con i corretti valori della distribuzione di massa di probabilità.*

La figura 4.7 confronta l'istogramma dei valori simulati dalla passeggiata casuale con l'effettiva distribuzione di probabilità `pd`. Si noti la somiglianza tra le due distribuzioni.

4.3.6 L'algoritmo di Metropolis

Vediamo ora come l'algoritmo di Metropolis possa venire usato per generare una catena di Markov irriducibile e aperiodica per la quale la distribuzione stazionaria è uguale alla distribuzione a posteriori di interesse.[7] In termini generali, l'algoritmo di Metropolis include due fasi.

- *Fase 1.* La selezione di un valore candidato θ' del parametro mediante il campionamento da una distribuzione proposta.
- *Fase 2.* La decisione tra la possibilità di accettare il valore candidato $\theta^{(m+1)} = \theta'$ o di mantenere il valore corrente $\theta^{(m+1)} = \theta$ sulla base del seguente criterio:
  - se $\mathcal{L}(\theta' \mid y)p(\theta') > \mathcal{L}(\theta \mid y)p(\theta)$ il valore candidato viene sempre accettato;
  - altrimenti il valore candidato viene accettato solo in una certa proporzione di casi.

Esaminiamo ora nei dettagli il funzionamento dell'algoritmo di Metropolis.

(a) Si inizia con un punto arbitrario $\theta^{(1)}$, quindi il primo valore della catena di Markov $\theta^{(1)}$ può corrispondere semplicemente ad un valore a caso tra i valori possibili del parametro.

(b) Per ogni passo successivo della catena, $m + 1$, si campiona un valore candidato θ' da una distribuzione proposta: $\theta' \sim \Pi(\theta)$. La distribuzione proposta può essere qualunque distribuzione, anche se, idealmente, è meglio che sia simile alla distribuzione a posteriori. In pratica, però, la distribuzione a posteriori è sconosciuta e quindi il valore θ' viene campionato da una qualche distribuzione simmetrica centrata sul valore corrente $\theta^{(m)}$ del parametro. Nell'esempio qui discusso, useremo la distribuzione gaussiana. Tale distribuzione sarà centrata sul valore corrente della catena e avrà una appropriata deviazione standard:

[7] Una illustrazione visiva di come si svolge il processo di "esplorazione" dell'algoritmo di Metropolis è fornita in questo post[8].

$\theta' \sim \mathcal{N}(\theta^{(m)}, \sigma)$. In pratica, questo significa che, se σ è piccola, il valore candidato θ' sarà simile al valore corrente $\theta^{(m)}$.

(c) Una volta generato il valore candidato θ' si calcola il rapporto tra la densità della distribuzione a posteriori non normalizzata nel punto θ' [ovvero, il prodotto tra la verosimiglianza $\mathcal{L}(y \mid \theta')$ nel punto θ' e la distribuzione a priori nel punto θ'] e la densità della distribuzione a posteriori non normalizzata nel punto $\theta^{(m)}$ [ovvero, il prodotto tra la verosimiglianza $\mathcal{L}(y \mid \theta^{(m)})$ nel punto $\theta^{(m)}$ e la distribuzione a priori nel punto $\theta^{(m)}$]:

$$\alpha = \frac{p(y \mid \theta')p(\theta')}{p(y \mid \theta^{(m)})p(\theta^{(m)})}. \tag{4.1}$$

Si noti che, essendo un rapporto, la (4.1) cancella la costante di normalizzazione.

(d) Il rapporto α viene utilizzato per decidere se accettare il valore candidato θ', oppure se campionare un diverso candidato. Possiamo pensare al rapporto α come alla risposta alla seguente domanda: alla luce dei dati, è più plausibile il valore candidato del parametro o il valore corrente? Se α è maggiore di 1 ciò significa che il valore candidato è più plausibile del valore corrente; in tali circostanze il valore candidato viene sempre accettato. Altrimenti, si decide di accettare il valore candidato con una probabilità minore di 1, ovvero non sempre, ma soltanto con una probabilità uguale ad α. Se α è uguale a 0.10, ad esempio, questo significa che la plausibilità a posteriori del valore candidato è 10 volte più piccola della plausibilità a posteriori del valore corrente. Dunque, il valore candidato verrà accettato solo nel 10% dei casi. Come conseguenza di questa strategia di scelta, l'algoritmo di Metropolis ottiene un campione casuale dalla distribuzione a posteriori, dato che la probabilità di accettare il valore candidato è proporzionale alla densità del candidato nella distribuzione a posteriori. Dal punto di vista algoritmico, la procedura descritta sopra viene implementata confrontando il rapporto α con un valore casuale estratto da una distribuzione uniforme Unif$(0, 1)$. Se $\alpha > u \sim$ Unif$(0, 1)$ allora il punto candidato θ' viene accettato e la catena si muove

in quella nuova posizione, ovvero $\theta^{(m+1)} = \theta'^{(m+1)}$. Altrimenti $\theta^{(m+1)} = \theta^{(m)}$ e si campiona un nuovo valore candidato θ'.

(e) Il passaggio finale dell'algoritmo calcola l'*accettanza* in una specifica esecuzione dell'algoritmo, ovvero la proporzione dei valori candidati θ' che sono stati accettati come valori successivi nella sequenza.

L'algoritmo di Metropolis prende come input il numero M di passi da simulare, la deviazione standard σ della distribuzione proposta e la densità a priori, e ritorna come output la sequenza $\theta^{(1)}, \theta^{(2)}, \dots, \theta^{(M)}$. La chiave del successo dell'algoritmo di Metropolis è il numero di passi fino a che la catena approssima la stazionarietà. Tipicamente i primi da 1000 a 5000 elementi sono scartati. Dopo un certo periodo k (detto di *burn-in*), la catena di Markov converge ad una variabile casuale che è distribuita secondo la distribuzione a posteriori. In altre parole, i campioni del vettore $\left(\theta^{(k+1)}, \theta^{(k+2)}, \dots, \theta^{(M)}\right)$ diventano campioni di $p(\theta \mid y)$.

4.3.7 Un esempio concreto (seconda versione)

Per fare un esempio concreto, consideriamo nuovamente i 30 pazienti esaminati da Zetsche et al. (2019). Di essi, 23 hanno manifestato aspettative distorte negativamente sul loro stato d'animo futuro. Utilizzando l'algoritmo di Metropolis, ci poniamo il problema di ottenere la stima a posteriori di θ (probabilità di manifestare un'aspettativa distorta negativamente), dati 23 "successi" in 30 prove, imponendo su θ la stessa distribuzione a priori usata nel Capitolo 2, ovvero Beta$(2, 10)$.

Per calcolare la funzione di verosimiglianza, avendo fissato i dati di Zetsche et al. (2019), definiamo la funzione `likelihood()`

```
likelihood <- function(param, x = 23, N = 30) {
  dbinom(x, N, param)
}
```

che ritorna l'ordinata della verosimiglianza binomiale per ciascun valore del vettore `param` in input.

La distribuzione a priori Beta$(2, 10)$ è implementata nella funzione `prior()`:

```
prior <- function(param, alpha = 2, beta = 10) {
  dbeta(param, alpha, beta)
}
```

Il prodotto della densità a priori e della verosimiglianza è implementato nella funzione `posterior()`:

```
posterior <- function(param) {
  likelihood(param) * prior(param)
}
```

L'Appendice **??** mostra come un'approssimazione della distribuzione a posteriori $p(\theta \mid y)$ per questi dati possa essere ottenuta mediante il metodo basato su griglia.

4.3.8 Implementazione

Per implementare l'algoritmo di Metropolis utilizzeremo una distribuzione proposta gaussiana. Il valore candidato sarà dunque un valore selezionato a caso da una gaussiana di parametri μ uguale al valore corrente nella catena e $\sigma = 0.9$. In questo esempio, la deviazione standard σ è stata scelta empiricamente in modo tale da ottenere una accettanza adeguata. L'accettanza ottimale è di circa 0.20 e 0.30 — se l'accettanza è troppo grande, l'algoritmo esplora uno spazio troppo ristretto della distribuzione a posteriori.[9]

```
proposal_distribution <- function(param) {
  while(1) {
    res = rnorm(1, mean = param, sd = 0.9)
    if (res > 0 & res < 1)
      break
  }
  res
}
```

[9] L'accettanza dipende dalla distribuzione proposta: in generale, tanto più la distribuzione proposta è simile alla distribuzione target, tanto più alta diventa l'accettanza.

Nella presente implementazione del campionamento dalla distribuzione proposta è stato inserito un controllo che impone al valore candidato di essere incluso nell'intervallo [0, 1].[10]

L'algoritmo di Metropolis viene implementato nella seguente funzione:

```
run_metropolis_MCMC <- function(startvalue, iterations) {
  chain <- vector(length = iterations + 1)
  chain[1] <- startvalue
  for (i in 1:iterations) {
    proposal <- proposal_distribution(chain[i])
    r <- posterior(proposal) / posterior(chain[i])
    if (runif(1) < r) {
      chain[i + 1] <- proposal
    } else {
      chain[i + 1] <- chain[i]
    }
  }
  chain
}
```

Avendo definito le funzioni precedenti, generiamo una catena di valori θ:

```
set.seed(123)
startvalue <- runif(1, 0, 1)
niter <- 1e4
chain <- run_metropolis_MCMC(startvalue, niter)
```

Mediante le istruzioni precedenti otteniamo una catena di Markov costituita da 10,001 valori. Escludiamo i primi 5,000 valori considerati come burn-in. Ci restano dunque con 5,001 valori che verranno considerati come un campione casuale estratto dalla distribuzione a posteriori $p(\theta \mid y)$.

L'accettanza è pari a

[10] Si possono trovare implementazioni dell'algoritmo di Metropolis più eleganti di quella presentata qui. Lo scopo dell'esercizio è quello di illustrare la logica soggiacente all'algoritmo di Metropolis, non quello di proporre un'implementazione efficente dell'algoritmo.

```
burnIn <- niter / 2
acceptance <- 1 - mean(duplicated(chain[-(1:burnIn)]))
acceptance
#> [1] 0.2511
```

il che conferma la bontà della deviazione standard ($\sigma = 0.9$) scelta per la distribuzione proposta.

A questo punto è facile ottenere una stima a posteriori del parametro θ. Per esempio, la stima della media a posteriori è:

```
mean(chain[-(1:burnIn)])
#> [1] 0.5922
```

Una figura che mostra l'approssimazione di $p(\theta \mid y)$ ottenuta con l'algoritmo di Metropolis, insieme ad un *trace plot* dei valori della catena di Markov, viene prodotta usando le seguenti istruzioni:

```
p1 <- tibble(
  x = chain[-(1:burnIn)]
) %>%
  ggplot(aes(x)) +
  geom_histogram() +
  labs(
    x = expression(theta),
    y = "Frequenza",
    title = "Distribuzione a posteriori"
  ) +
  geom_vline(
    xintercept = mean(chain[-(1:burnIn)])
  )
p2 <- tibble(
  x = 1:length(chain[-(1:burnIn)]),
  y = chain[-(1:burnIn)]
) %>%
  ggplot(aes(x, y)) +
  geom_line() +
  labs(
```

```
    x = "Numero di passi",
    y = expression(theta),
    title = "Valori della catena"
  ) +
  geom_hline(
    yintercept = mean(chain[-(1:burnIn)]),
    colour = "gray"
  )
p1 + p2
```

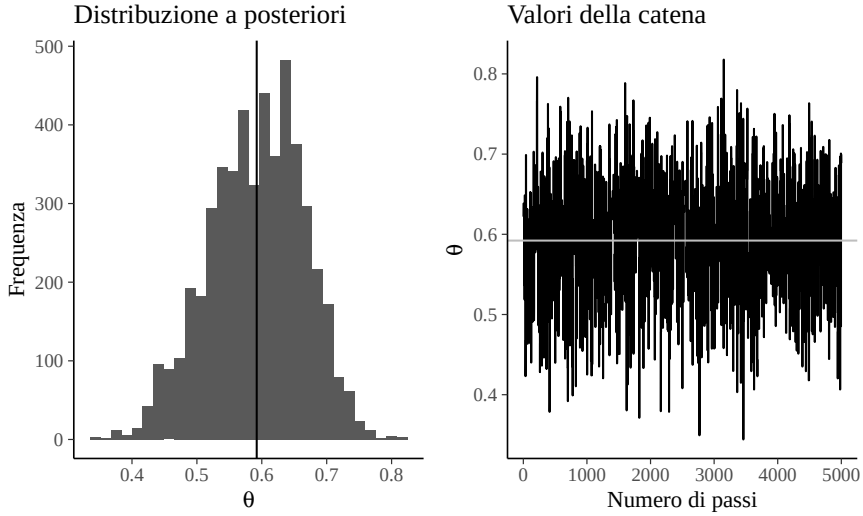


Figura 4.8: *Sinistra. Stima della distribuzione a posteriori della probabilità di una aspettativa futura distorta negativamente per i dati di Zetsche et al. (2019). Destra. Trace plot dei valori della catena di Markov escludendo il periodo di burn-in.*

4.3.8.1 Input

Negli esempi discussi in questo Capitolo abbiamo illustrato l'esecuzione di una singola catena in cui si parte un unico valore iniziale e si raccolgono i valori simulati da molte iterazioni. È possibile che i valori di una catena siano influenzati dalla scelta del valore iniziale. Quindi una raccomandazione generale è di eseguire l'algoritmo di Metropolis più volte utilizzando diversi valori di partenza. In questo caso, si avranno

più catene di Markov. Confrontando le proprietà delle diverse catene si esplora la sensibilità dell'inferenza alla scelta del valore di partenza. I software MCMC consentono sempre all'utente di specificare diversi valori di partenza e di generare molteplici catene di Markov.

4.3.8.2 Stazionarietà

Un punto importante da verificare è se il campionatore ha raggiunto la sua distribuzione stazionaria. La convergenza di una catena di Markov alla distribuzione stazionaria viene detta "mixing".

4.3.8.3 Autocorrelazione

Informazioni sul "mixing" della catena di Markov sono fornite dall'autocorrelazione. L'autocorrelazione misura la correlazione tra i valori successivi di una catena di Markov. Il valore m-esimo della serie ordinata viene confrontato con un altro valore ritardato di una quantità k (dove k è l'entità del ritardo) per verificare quanto si correli al variare di k. L'autocorrelazione di ordine 1 (*lag 1*) misura la correlazione tra valori successivi della catena di Markow (cioè, la correlazione tra $\theta^{(m)}$ e $\theta^{(m-1)}$); l'autocorrelazione di ordine 2 (*lag 2*) misura la correlazione tra valori della catena di Markow separati da due "passi" (cioè, la correlazione tra $\theta^{(m)}$ e $\theta^{(m-2)}$); e così via.

L'autocorrelazione di ordine k è data da ρ_k e può essere stimata come:

$$\begin{aligned}\rho_k &= \frac{\text{Cov}(\theta_m, \theta_{m+k})}{\mathbb{V}(\theta_m)} \\ &= \frac{\sum_{m=1}^{n-k}(\theta_m - \bar{\theta})(\theta_{m-k} - \bar{\theta})}{\sum_{m=1}^{n-k}(\theta_m - \bar{\theta})^2} \qquad \text{con} \quad \bar{\theta} = \frac{1}{n}\sum_{m=1}^{n}\theta_m. \end{aligned} \tag{4.2}$$

Per fare un esempio pratico, simuliamo dei dati autocorrelati con la funzione R `colorednoise::colored_noise()`:

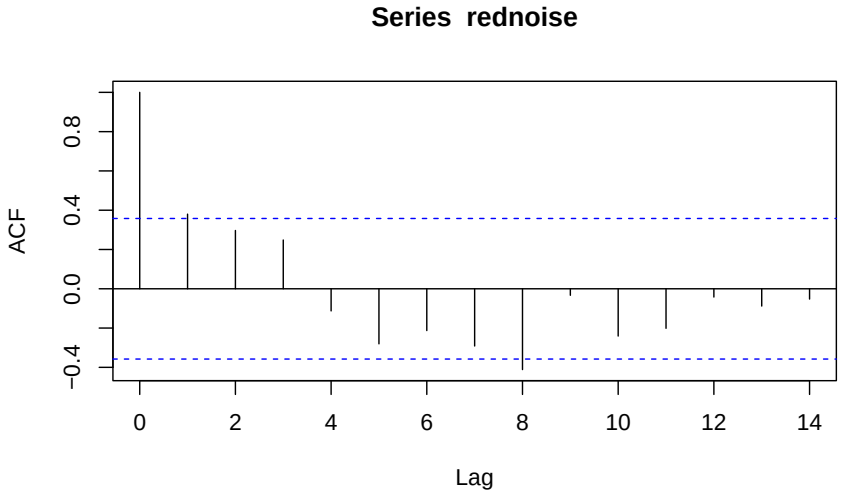
```
suppressPackageStartupMessages(library("colorednoise"))
set.seed(34783859)
rednoise <- colored_noise(
  timesteps = 30, mean = 0.5, sd = 0.05, phi = 0.3
)
```

L'autocorrelazione di ordine 1 è semplicemente la correlazione tra ciascun elemento e quello successivo nella sequenza. Nell'esempio, il vettore `rednoise` è una sequenza temporale di 30 elementi. Il vettore `rednoise[-length(rednoise)]` include gli elementi con gli indici da 1 a 29 nella sequenza originaria, mentre il vettore `rednoise[-1]` include gli elementi 2:30. Gli elementi delle coppie ordinate dei due vettori avranno dunque gli indici $(1, 2), (2, 3), ... (29, 30)$ degli elementi della sequenza originaria. La correlazione di Pearson tra i vettori `rednoise[-length(rednoise)]` e `rednoise[-1]` corrisponde dunque all'autocorrelazione di ordine 1 della serie temporale.

```
cor(rednoise[-length(rednoise)], rednoise[-1])
#> [1] 0.3967
```

Il Correlogramma è uno strumento grafico usato per la valutazione della tendenza di una catena di Markov nel tempo. Il correlogramma si costruisce a partire dall'autocorrelazione ρ_k di una catena di Markov in funzione del ritardo (*lag*) k con cui l'autocorrelazione è calcolata: nel grafico ogni barretta verticale riporta il valore dell'autocorrelazione (sull'asse delle ordinate) in funzione del ritardo (sull'asse delle ascisse). In R, il correlogramma può essere prodotto con una chiamata a `acf()`:

```
acf(rednoise)
```



Il correlogramma precedente mostra come l'autocorrelazione di ordine 1 sia circa pari a 0.4 e diminuisce per lag maggiori; per lag di 4, l'autocorrelazione diventa negativa e aumenta progressivamente fino ad un lag di 8; eccetera.

In situazioni ottimali l'autocorrelazione diminuisce rapidamente ed è effettivamente pari a 0 per piccoli lag. Ciò indica che i valori della catena di Markov che si trovano a più di soli pochi passi di distanza gli uni dagli altri non risultano associati tra loro, il che fornisce conferma del "mixing" della catena di Markov, ossia di convergenza alla distribuzione stazionaria. Nelle analisi bayesiane, una delle strategie che consentono di ridurre l'autocorrelazione è quella di assottigliare l'output immagazzinando solo ogni m-esimo punto dopo il periodo di burn-in. Una tale strategia va sotto il nome di *thinning*.

4.3.8.4 Test di convergenza

Un test di convergenza può essere svolto in maniera grafica mediante le tracce delle serie temporali (*trace plot*), cioè il grafico dei valori simulati rispetto al numero di iterazioni. Se la catena è in uno stato stazionario le tracce mostrano assenza di periodicità nel tempo e ampiezza costante, senza tendenze visibili o andamenti degni di nota. Un esempio di *trace plot* è fornito nella figura 4.8 (destra).

Ci sono inoltre alcuni test che permettono di verificare la stazionarietà del campionatore dopo un dato punto. Uno è il test di Geweke che suddivide il campione, dopo aver rimosso un periodo di burn in, in due parti. Se la catena è in uno stato stazionario, le medie dei due campioni dovrebbero essere uguali. Un test modificato, chiamato Geweke z-score, utilizza un test z per confrontare i due subcampioni ed il risultante test statistico, se ad esempio è più alto di 2, indica che la media della serie sta ancora muovendosi da un punto ad un altro e quindi è necessario un periodo di burn-in più lungo.

Commenti e considerazioni finali

In generale, la distribuzione a posteriori dei parametri di un modello statistico non può essere determinata per via analitica. Tale problema viene invece affrontato facendo ricorso ad una classe di algoritmi per il

campionamento da distribuzioni di probabilità che sono estremamente onerosi dal punto di vista computazionale e che possono essere utilizzati nelle applicazioni pratiche solo grazie alla potenza di calcolo dei moderni computer. Lo sviluppo di software che rendono sempre più semplice l'uso dei metodi MCMC, insieme all'incremento della potenza di calcolo dei computer, ha contribuito a rendere sempre più popolare il metodo dell'inferenza bayesiana che, in questo modo, può essere estesa a problemi di qualunque grado di complessità.

Bibliografia

Albert, J. and Hu, J. (2019). *Probability and Bayesian Modeling*. Chapman and Hall/CRC.

Bechdel, A. (1986). *Dykes to watch out for*. Firebrand Books.

Horn, S. and Loewenstein, G. (2021). Underestimating learning by doing. *Available at SSRN 3941441*.

Johnson, A. A., Ott, M., and Dogucu, M. (2022). *Bayes Rules! An Introduction to Bayesian Modeling with R*. CRC Press.

Kruschke, J. (2014). *Doing Bayesian data analysis: A tutorial with R, JAGS, and Stan*. Academic Press.

Lunn, D., Jackson, C., Best, N., Thomas, A., and Spiegelhalter, D. (2013). The bugs book. *A Practical Introduction to Bayesian Analysis, Chapman Hall, London*.

Martin, O. A., Kumar, R., and Lao, J. (2022). *Bayesian Modeling and Computation in Python*. CRC Press.

McElreath, R. (2020). *Statistical rethinking: A Bayesian course with examples in R and Stan*. CRC Press, Boca Raton, Florida, 2nd edition edition.

Milgram, S. (1963). Behavioral study of obedience. *The Journal of Abnormal and Social Psychology*, 67(4):371–378.

van de Schoot, R., Depaoli, S., King, R., Kramer, B., Märtens, K., Tadesse, M. G., Vannucci, M., Gelman, A., Veen, D., Willemsen, J., and Yau, C. (2021). Bayesian statistics and modelling. *Nature Reviews Methods Primer*, 1(1):1–26.

Zetsche, U., Bürkner, P.-C., and Renneberg, B. (2019). Future expectations in clinical depression: Biased or realistic? *Journal of Abnormal Psychology*, 128(7):678–688.