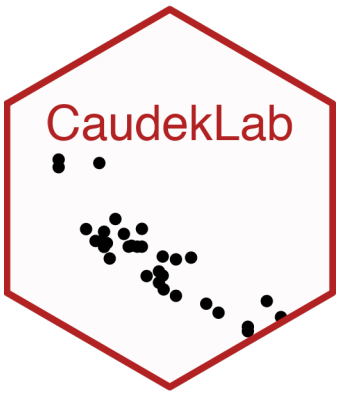
Psicometria

Corrado Caudek

2021-11-11

Questo documento è stato realizzato con:

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



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

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

Indice

Data della versione presente: Novembre 11, 2021.

Prefazione

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

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

La psicologia e la Data Science

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

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

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

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

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

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

Come studiare

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

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

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

Sviluppare un metodo di studio efficace

> Memorization is not learning. — Richard Phillips Feynman

Avendo insegnato 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 (Burger & Starbird, 2012).

- 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 possiamo solo applicare qualcosa che già sappiamo fare. Ma se dobbiamo imparare qualcosa di nuovo, la pressione è un impedimento.

- È utile farsi da soli delle domande sugli argomenti trattati, senza limitarsi a cercare di risolvere gli esercizi che vengono assegnati. Quando studio qualcosa mi viene in mente: "se questo è vero, allora deve succedere quest'altra cosa". Allora verifico se questo è vero, di solito con una simulazione. Se i risultati della simulazione sono quelli che mi aspetto, allora vuol dire che ho capito. Se i risultati sono diversi da quelli che mi aspettavo, allora mi rendo conto di non avere capito e ritorno indietro a studiare con più attenzione la teoria che pensavo di avere capito – e ovviamente mi rendo conto che c'era un aspetto che avevo frainteso. Questo tipo di verifica è qualcosa che dobbiamo fare da soli, in prima persona: nessun altro può fare questo al posto nostro.

- Non aspettatevi di capire tutto la prima volta che incontrate un argomento nuovo.[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.

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

- Tutti noi dobbiamo imparare l'arte di trovare le informazioni, non solo nel caso di questo insegnamento. Quando vi trovate di fronte a qualcosa che non capite, o ottenete un oscuro messaggio di errore da un software, ricordatevi: "Google is your friend".

Corrado Caudek

Febbraio 2022

Capitolo 1

Valutare e confrontare i modelli

In breve

Il principio base del metodo scientifico è la *replicabilità* delle osservazioni: le osservazioni che non possono essere replicate sono poco interessanti. Parallelamente, una caratteristica fondamentale di un modello scientifico è la *generalizzabilità*: se un modello è capace di descrivere soltanto le proprietà di uno specifico campione di osservazioni, allora è poco utile. Ma come è possibile valutare la generalizzabilità di un modello statistico? Questa è la domanda a cui cercheremo di rispondere in questo Capitolo.

Nel valutare un modello, il ricercatore deve porsi tre domande critiche.

- Quali conseguenze più ampie derivano dall'inferenza? Come e chi ha raccolto i dati? Colui che svolge la ricerca otterrebbe di benefici manipolando i dati (escludendo delle osservazioni; selezionando il campione)? Che impatto hanno inferenze che vengono tratte dai dati sugli individui e sulla società? Quali pregiudizi o strutture di potere possono essere coinvolti in questa analisi?
- Che tipo di distorsioni sistematiche potrebbero essere presenti nell'analisi statistica? Ricordiamo la famosa citazione di George Box: "Tutti i modelli sono sbagliati, ma alcuni sono utili". È dunque importante sapere quanto è sbagliato il modello. Le assunzioni che stanno alla base del modello sono ragionevoli? Il meccanismo generatore dei dati che è stato ipotizzato è adeguato per il fenomeno in esame?
- Quanto è accurato il modello? Quanto sono lontane dalla realtà le previsioni del modello?

Per approfondire questi temi, si rinvia al testo di Johnson et al. (2022). Qui ci concentreremo sul tema della generalizzabilità del modello.

1.1 Capacità predittiva

Nel framework bayesiano il problema della generalizzabilità del modello viene affrontato calcolando la capacità predittiva del modello, laddove per capacità predittiva si intende la capacità di un modello, i cui parametri sono stati stimati usando le informazioni di un campione, di ben adattarsi ad un campione di nuove osservazioni future. A questo proposito, in questo Capitolo cercheremo di rispondere a due tipi di domande.

1. Quali criteri devono essere considerati se vogliamo valutare la capacità predittiva di un modello?
2. Come è possibile misurare la capacità predittiva di un modello sulla base delle informazioni di un campione di osservazioni? E, parallelamente, come possiamo confrontare la capacità predittiva di modelli alternativi?

1.2 Il rasoio di Ockham

Uno dei problemi più importanti in campo scientifico è quello di scegliere il modello più adatto a spiegare un insieme di informazioni. Le domande che i ricercatori si pongono sono: il modello è completo? È necessario un nuovo parametro? Come si può cambiare il modello? Se ci sono delle alternative, quali sono le migliori?

Per rispondere a tali domandi, ci viene in soccorso il principio del rasoio di Ockham: *frustra fit per plura quod potest fieri per pauciora* ("si fa inutilmente con molte cose ciò che si può fare con poche cose"). Parafrasando la massima potremmo dire che: quando due modelli approssimano egualmente bene le osservazioni viene sempre preferito quello più semplice. Questo principio è alla base della scienza e, da un punto di vista della probabilità, il fatto che il rasoio di Ockham funzioni è data dal fatto che i modelli più semplici sono anche i più probabili.

Due modelli possono fare le stesse predizioni ma possono differire in termini di complessità — per esempio, relativamente al numero di parametri richiesti. In questo caso, è facile decidere perché, in generale, i ricercatori preferiscono sempre i modelli più semplici (pragmaticamente, sono i più facili da usare). Tuttavia, il rasoio di Ockham, se usato da solo, non ci consente sempre di selezionare tra modelli alternativi, perché i modelli possono differire sia per complessità (ovvero, per il numero di parametri) sia per accuratezza (ovvero, per la grandezza degli errori di predizione). Il rasoio di Ockham non è in grado di trovare un equilibrio tra la necessità di aumentare l'accuratezza e quella di diminuire la complessità.

In questo Capitolo ci chiederemo come sia possibile misurare l'accuratezza predittiva di un modello. Ciò ci consentirà, in seguito, di usare il rasoio di Ockham: a parità di accuratezza, verrà scelto il modello più semplice. Ma nella pratica scientifica non si sacrifica mai l'accuratezza per la semplicità: il criterio prioritario è sempre l'accuratezza.

Per valutare l'accuratezza (predittiva) di un modello, McElreath (2020) fa notare che è necessario evitare due opposti errori:

Tabella 1.1: *Previsioni meteo del primo annunciatore (P = pioggia, S = sole).*

giorno	1	2	3	4	5	6	7	8	9	10
previsione	1.0	1.0	1.0	0.6	0.6	0.6	0.6	0.6	0.6	0.6
osservazione	P	P	P	S	S	S	S	S	S	S

- *sovra-adattamento* (*overfitting*): il modello non si generalizza bene a nuovi dati futuri perché rappresenta gli aspetti idiosincratici o le informazioni irrilevanti che sono presenti nello specifico campione esaminato che, necessariamente, non si ritroveranno in altri campioni;
- *sotto-adattamento* (*underfitting*): il modello non è in grado di rendere conto delle regolarità presenti nei dati osservati.

In questo capitolo verranno presentati due metodi [detti *criterio di informazione* (*information criteria*) e *validazione incrociata* (*cross-validation*)] che consentono di valutare la capacità predittiva di un modello e di determinare quale, tra due o più modelli, sia quello preferibile.

Stargazing

Nella pratica concreta della ricerca, il metodo più comune per la selezione tra vari modelli statistici fa leva sui test di ipotesi statistiche di stampo frequentista. Questo metodo viene chiamato *stargazing*, poiché richiede soltanto l'esame degli asterischi (∗∗) che si trovano nell'output di un software statistico (gli asterischi marcano i coefficienti del modello che sono "statisticamente significativi"): alcuni ricercatori ritengono che il modello con più stelline sia il modello migliore. Ma questo non è vero. Al di là dei problemi legati ai test dell'ipotesi nulla, è sicuramente un errore usare i test di significatività per la selezione di modelli: i valori-p non consentono di trovare un equilibrio tra *underfitting* e *overfitting*. Infatti, variabili che migliorano la capacità predittiva di un modello non sempre sono statisticamente significative; inoltre, variabili statisticamente significative non sempre migliorano la capacità predittiva di un modello.

1.3 L'incertezza della previsione

Nella teoria dell'informazione, i criteri usati per definire l'accuratezza predittiva di un modello sono chiamati *target*. Tuttavia, un target può essere definito in molti modi diversi. Per spiegare questo punto, McElreath (2020) fa l'esempio di due annunciatori che danno il notiziario meteorologico alla TV per 10 giorni consecutivi. Le previsioni del primo annunciatore (formulate nei termini della *probabilità di pioggia*) sono riportate nella tabella 1.1; il secondo annunciatore prevede sempre sole (tabella 1.2). Il problema che ci poniamo è di stabilire quale annunciatore sia più accurato.

Un primo possibile criterio per scegliere tra i due annunciatori può essere formulato nei termini del numero medio di previsioni corrette. In base a tale criterio, il primo annunciatore ottiene un numero medio di successi uguale a $(3 \cdot 1 + 7 \cdot 0.4)/10 = 0.58$ mentre il secondo annunciatore ottiene un numero

Tabella 1.2: *Previsioni meteo del secondo annunciatore (P = pioggia, S = sole).*

giorno	1	2	3	4	5	6	7	8	9	10
previsione	0	0	0	0	0	0	0	0	0	0
osservazione	P	P	P	S	S	S	S	S	S	S

Tabella 1.3: *Analisi costi-benefici. P = pioggia, S = sole*

giorno	1	2	3	4	5	6	7	8	9	10
osservazione	P	P	P	S	S	S	S	S	S	S
annunciatore_1	-1.0	-1.0	-1.0	-0.6	-0.6	-0.6	-0.6	-0.6	-0.6	-0.6
annunciatore_2	-5	-5	-5	0	0	0	0	0	0	0

medio di successi uguale a $(3 \cdot 0 + 7 \cdot 1)/10 = 0.7$. In base a questo criterio, dunque, il secondo annunciatore è più accurato del primo.

Ma il numero medio di previsioni corrette non è l'unico criterio possibile per valutare l'accuratezza. Un secondo possibile criterio è un'analisi costi-benefici. Sia -1 il costo di portare con sé un ombrello e -5 il costo di bagnarsi.

Consideriamo la tabella 1.3. Per il primo annunciatore, quando la previsione è pioggia e effettivamente piove, il costo è -1; quando la previsione è pioggia al 60% e invece c'è il sole, il costo è -0.6 (si porta l'ombrello nel 60% dei casi). Per il primo annunciatore, dunque, il costo totale è $3 \cdot (-1) + 7 \cdot (-0.6) = -7.2$. Per il secondo annunciatore, invece, il costo totale è $3 \cdot (-5) + 7 \cdot (0) = -15$. Nei termini di un'analisi costi-benefici, dunque, il primo annunciatore fornisce previsioni più accurate (ovvero, predizioni a cui è associato un costo minore) del secondo.

È possibile cambiare la regola decisionale (-1, -5), ma sicuramente il costo di bagnarsi è molto maggiore del costo di portare con sé un ombrello. Quindi, quale che sia la regola decisionale, il fatto di trovandosi sempre impreparato quando piove pone il secondo annunciatore in una posizione di svantaggio rispetto al primo annunciatore.

Dalle considerazioni precedenti McElreath (2020) trae due conclusioni: (a) un'analisi costi-benefici non fornisce una misura univoca dell'accuratezza di un modello (ovvero, sono possibili diverse regole decisionali) e (b) il numero di successi non fornisce una metrica migliore di un'analisi costi-benefici. Non è dunque ancora chiaro quale sia il modo migliore per definire l'accuratezza di un modello.

1.4 Misurare l'accuratezza

McElreath (2020) sostiene che la misura migliore dell'accuratezza predittiva di un modello sia fornita dalla massimizzazione della funzione che descrive il vero processo generatore dei dati. Per spiegare cosa significa questo, chiediamoci quale sia la probabilità di osservare la *specifica* sequenza di sole e pioggia (ovvero, P, P, P, S, S, S, S, S, S, S) che i due annunciatori vogliono prevedere. La probabilità cercata è data dal prodotto della probabilità di prevedere correttamente ($y = 1$) le condizioni meteo in ciascuno dei 10 giorni considerati. In altre parole, la probabilità cercata è la verosimiglianza congiunta dei dati. In termini più precisi,

se $y = y_1, \dots, y_{10}$ sono i dati e se θ è la probabilità dell'evento $y_i = 1$, allora $p(y \mid \theta) = \prod_{i=1}^{10} p(y_i \mid \theta)$. Per il primo annunciatore la verosimiglianza congiunta è $1^3 \cdot 0.4^7 \approx 0.005$; per il secondo annunciatore è $0^3 \cdot 1^7 = 0$. Vi è dunque una probabilità pari a zero che il secondo annunciatore predica la sequenza corretta di sole e pioggia nei 10 giorni considerati (infatti, il secondo annunciatore non predice mai pioggia). In conclusione, benché per il secondo annunciatore la probabilità *media* di previsioni corrette (numero medio di successi) sia alta, la probabilità *congiunta* di previsioni corrette è zero.

Regola di punteggio logaritmica

Nella letteratura statistica la probabilità congiunta viene utilizzata quale misura della precisione delle previsioni probabilistiche ed è chiamata *regola di punteggio* (*scoring rule*). Nel calcolo della regola di punteggio si utilizzano dei pesi che vengono ottimizzati in modo tale che il *punteggio della previsione* (*score*) sia il più elevato possibile. Il punteggio della previsione è solitamente espresso su scala logaritmica (*log score*). Una tale *regola di punteggio logaritmica* è importante nel confronto di modelli statistici a causa della sua connessione con la misura della divergenza di Kullback-Leibler.

Esempio 1.1. Nell'esempio dei due annunciatori, esaminiamo la previsione meteo relativa ad un giorno soltanto e ipotizziamo che l'annunciatore attribuisca all'evento pioggia una probabilità dell'60% (e quindi sole avrà una probabilità del 40%). Se la previsione si rivela corretta, l'annunciatore avrà un log score di -0.51 [calcolato come: *log score* $= \log(0.6) = -0.51$]; se invece si rivela falsa lo score sarà -0.92 [calcolato come: *log score* $= \log(0.4) = -0.92$].

1.5 Entropia

Per discutere la divergenza di Kullback-Leibler è prima necessario introdurre il concetto di entropia. La nozione di entropia fu introdotta agli inizi del XIX secolo nel campo della termodinamica classica; il secondo principio della termodinamica è infatti basato sul concetto di entropia che, in generale, è assunto come una misura del disordine di un sistema fisico. Successivamente Boltzmann fornì una definizione statistica di entropia. Nel 1948 Shannon impiegò poi la nozione di entropia nell'ambito della teoria delle comunicazioni. Si giunge così alla seguente definizione.

Definizione 1.1. Sia $Y = y_1, \dots, y_n$ una variabile casuale e $p_t(y)$ una distribuzione di probabilità su Y. Si definisce la sua entropia (detta di Shannon) come:

$$H(Y) = -\sum_{i=1}^{n} p_t(y_i) \cdot \log p_t(y_i). \tag{1.1}$$

Intuitivamente, l'entropia quantifica la riduzione del nostro grado di incertezza relativa ad un evento una volta che l'esito dell'evento è stato appreso.[1]

[1] È possibile pensare all'entropia nei termini del numero di domande sì/no che devono essere

Proprietà

Si possono evidenziare due proprietà dell'entropia.

- L'entropia aumenta all'aumentare della varianza di una variabile casuale.
- L'entropia aumenta all'aumentare del numero di possibilità con cui un evento può verificarsi.

Esempio 1.2. Consideriamo il lancio di una moneta, non necessariamente bilanciata, dove sono note le probabilità che esca "testa" oppure "croce". L'entropia dell'esito del prossimo lancio è massima se la moneta non è truccata, cioè quando "testa" e "croce" hanno la stessa probabilità (pari a 1/2) di verificarsi. Questa è infatti la situazione di massima incertezza, ovvero, in cui è più difficile prevedere se uscirà "testa" oppure "croce". D'altra parte, quando la moneta è truccata uno degli esiti avrà una probabilità maggiore di verificarsi e quindi c'è meno incertezza, il che si riflette in una minore entropia. Il caso estremo è quello in cui la moneta ha lo stesso simbolo su entrambe le facce: in questo caso non c'è incertezza e l'entropia è zero.

Esempio 1.3. Un altro esempio riguarda le previsioni del tempo. Supponiamo che le vere probabilità di pioggia e sole siano, rispettivamente, $p_1 = 0.3$ e $p_2 = 0.7$. Quindi

$$H(p) = -[p(y_1) \log p(y_1) + p(y_2) \log p(y_2)] \approx 0.61.$$

Svolgendo i calcoli in R abbiamo:

```
p <- c(0.3, 0.7)
-sum(p * log(p))
#> [1] 0.611
```

Se però viviamo a Las Vegas, allora le probabilità di pioggia e sole saranno qualcosa come $p(y_1) = 0.01$ e $p(y_2) = 0.99$. In questo secondo caso, l'entropia è 0.06, ovvero, molto minore di prima. Infatti, a Las Vegas non piove quasi mai, per cui quando abbiamo imparato che, in un certo giorno, non ha piovuto, abbiamo imparato molto poco rispetto a quello che sapevamo in precedenza.

Esempio 1.4. Abbiamo visto in precedenza che, se gli esiti possibili sono pioggia o sole con $p(y_1) = 0.7$, $p(y_2) = 0.3$, allora l'entropia è

```
-(0.7 * log(0.7) + 0.3 * log(0.3))
#> [1] 0.611
```

poste per ridurre l'incertezza. Per esempio, se in un certo giorno ci può essere solo sole o pioggia, per ridurre l'incertezza, a fine giornata chiediamo: "ha piovuto?" La risposta (sì/no) ad una singola domanda elimina l'incertezza, e quindi l'informazione ottenuta (ovvero, la riduzione dell'incertezza) è uguale ad 1 bit. Se in una certa giornata ci potrebbero essere sole, pioggia o neve, per ridurre l'incertezza sono necessarie due domande: "c'era sole?"; "ha piovuto?" In questo secondo caso, l'informazione ottenuta (ovvero, la riduzione dell'incertezza) è uguale ad 2 bit. Usando un logaritmo in base 2, dunque, l'entropia può essere interpretata come il numero minimo di bit necessari per codificare la quantità di informazione nei dati.

Se gli esiti possibili sono pioggia, neve o sole con $p(y_1) = 0.7$, $p(y_2) = 0.15$ e $p(y_3) = 0.15$, rispettivamente, allora l'entropia è maggiore, ovvero è pari a 0.82.

```
-(0.7 * log(0.7) + 0.15 * log(0.15) + 0.15 * log(0.15))
#> [1] 0.819
```

1.6 Dall'entropia all'accuratezza

Anche se il valore assoluto dell'entropia è difficile da interpretare, nel seguito vedremo come l'entropia possa essere usata per misurare l'accuratezza di un modello statistico. Nello specifico, ci porremo il problema di misurare la distanza tra la distribuzione di probabilità ipotizzata da un modello, chiamiamola $p_{\mathcal{M}}$, e la distribuzione di probabilità del vero modello generatore dei dati, p_t. Nella teoria delle probabilità e nella teoria dell'informazione, la misura dell'informazione persa quando $p_{\mathcal{M}}$ è usata per approssimare p_t viene detta *divergenza di Kullback-Leibler*. La divergenza di Kullback-Leibler, denotata con $D_{KL}(p_t \,||\, p_{\mathcal{M}})$, misura l'incremento della nostra incertezza quando una distribuzione "approssimata" viene usata al posto della "vera" distribuzione di probabilità.

Definizione 1.2. Per due distribuzioni discrete p_t e $p_{\mathcal{M}}$, la divergenza KL di $p_{\mathcal{M}}$ da p_t è definita come:

$$D_{KL}(p_t \,||\, p_{\mathcal{M}}) = \sum_{i=1}^{n} p_t(y_i) \cdot [\log p_t(y_i) - \log p_{\mathcal{M}}(y_i)] \,. \tag{1.2}$$

La divergenza di Kullback-Leibler introduce un piccolo cambiamento alla (1.1): anziché considerare una sola distribuzione di probabilità, p_t, considera anche una approssimazione a tale distribuzione, ovvero $p_{\mathcal{M}}$. Calcolando la differenza dei logaritmi dei valore delle due distribuzioni giungiamo alla (1.2).

La divergenza KL misura la quantità di informazione che viene persa quando una distribuzione approssimata viene usata per descrivere le proprietà della distribuzione di riferimento. Infatti, ciò che viene calcolato nella divergenza KL è il valore atteso della differenza tra il logaritmo delle probabilità dei dati nella distribuzione p_t e il logaritmo delle probabilità dei dati nella distribuzione approssimata $p_{\mathcal{M}}$:

$$D_{KL}(p_t \,||\, p_{\mathcal{M}}) = \mathbb{E}\left[\log p_t(y) - \log p_{\mathcal{M}}(y)\right] . \tag{1.3}$$

Se c'è una perfetta corrispondenza tra le due distribuzioni, $p_t = p_{\mathcal{M}}$, allora

$$D_{KL}(p_t \,||\, p_{\mathcal{M}}) = D_{KL}(p_t \,||\, p_t) = \sum_{i=1}^{n} p_t(y_i) \cdot [\log p_t(y_i) - \log p_t(y_i)] = 0,$$

nessuna incertezza aggiuntiva viene introdotta se una distribuzione viene usata per rappresentare se stessa. Altrimenti (ovvero, se $p_t \neq p_{\mathcal{M}}$) la divergenza KL assume valori nell'intervallo $[0, \infty]$: all'aumentare della differenza tra $p_{\mathcal{M}}$ e p_t aumenta anche il valore $D_{KL}(p_t \,||\, p_{\mathcal{M}})$.

Esempio 1.5. (da McElreath, 2020) Sia la distribuzione target $p = \{0.3, 0.7\}$. Supponiamo che la distribuzione approssimata q possa assumere valori da $q = \{0.01, 0.99\}$ a $q = \{0.99, 0.01\}$. Calcoliamo la divergenza KL.

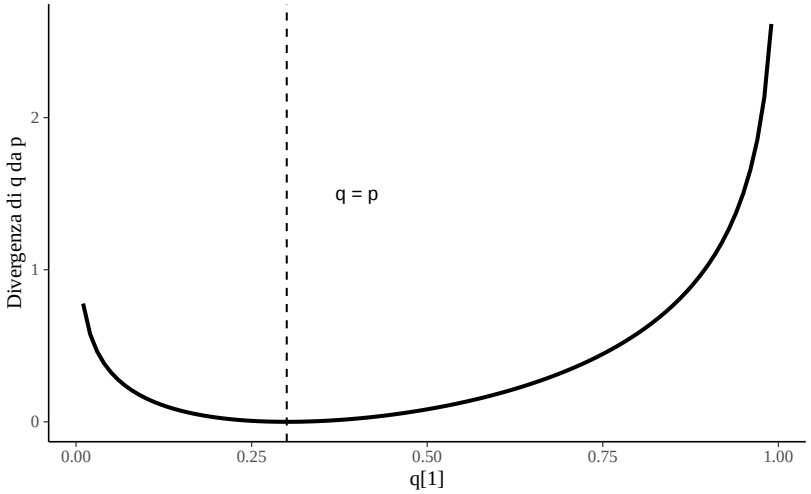
Le istruzioni R sono le seguenti:

```
t <-
  tibble(
    p_1 = .3,
    p_2 = .7,
    q_1 = seq(from = .01, to = .99, by = .01)
  ) %>%
  mutate(
    q_2 = 1 - q_1
  ) %>%
  mutate(
    d_kl = (p_1 * log(p_1 / q_1)) + (p_2 * log(p_2 / q_2))
  )

head(t)
#> # A tibble: 6 x 5
#>     p_1   p_2   q_1   q_2  d_kl
#>   <dbl> <dbl> <dbl> <dbl> <dbl>
#> 1   0.3   0.7  0.01  0.99 0.778
#> 2   0.3   0.7  0.02  0.98 0.577
#> 3   0.3   0.7  0.03  0.97 0.462
#> 4   0.3   0.7  0.04  0.96 0.383
#> 5   0.3   0.7  0.05  0.95 0.324
#> 6   0.3   0.7  0.06  0.94 0.276
```

Nella figura seguente sull'asse delle ascisse sono rappresentati i valori q e sull'asse delle ordinante sono riportati i corrispondenti valori D_{KL}.

```
t %>%
  ggplot(aes(x = q_1, y = d_kl)) +
  geom_vline(xintercept = .3, linetype = 2) +
  geom_line(size = 1) +
  annotate(
    geom = "text", x = .4, y = 1.5, label = "q = p",
    size = 3.5
  ) +
  labs(
    x = "q[1]",
    y = "Divergenza di q da p"
  )
```



Tanto meglio la distribuzione q approssima la distribuzione target tanto più piccolo è il valore di divergenza KL.

Esempio 1.6. Sia p una distribuzione binomiale di parametri $\theta = 0.2$ e $n = 5$

```
n <- 4
p <- 0.2
true_py <- dbinom(0:n, n, 0.2)
true_py
#> [1] 0.4096 0.4096 0.1536 0.0256 0.0016
```

e sia q_1 una approssimazione a p:

```
q1 <- c(0.46, 0.42, 0.10, 0.01, 0.01)
q1
#> [1] 0.46 0.42 0.10 0.01 0.01
```

Sia q_2 una distribuzione uniforme:

```
q2 <- rep(0.2, 5)
q2
#> [1] 0.2 0.2 0.2 0.2 0.2
```

La divergenza KL di q_1 da p è

```
sum(true_py * log(true_py / q1))
#> [1] 0.0293
```

La divergenza KL di q_2 da p è:

```
sum(true_py * log(true_py / q2))
#> [1] 0.486
```

È chiaro che perdiamo una quantità maggiore di informazioni se, per descrivere la distribuzione binomiale p, usiamo la distribuzione uniforme q_2 anziché q_1.

La divergenza dipende dalla direzione

La divergenza KL non è simmetrica: la KL da p_t a $p_{\mathcal{M}}$ in generale è diversa dalla KL da $p_{\mathcal{M}}$ a p_t.

Esempio 1.7. Usando le seguenti istruzioni R otteniamo:

```
tibble(
  direction = c("Da q a p", "Da p a q"),
  p_1 = c(.01, .7),
  q_1 = c(.7, .01)
) %>%
  mutate(
    p_2 = 1 - p_1,
    q_2 = 1 - q_1
  ) %>%
  mutate(d_kl = (p_1 * log(p_1 / q_1)) + (p_2 * log(p_2 / q_2)))
#> # A tibble: 2 x 6
#>   direction   p_1   q_1   p_2   q_2  d_kl
#>   <chr>     <dbl> <dbl> <dbl> <dbl> <dbl>
#> 1 Da q a p   0.01  0.7   0.99  0.3   1.14
#> 2 Da p a q   0.7   0.01  0.3   0.99  2.62
```

1.7 Expected log predictive density

Nel caso continuo, la divergenza KL diventa:

$$D_{KL}(p_t \,||\, p_{\mathcal{M}}) = \int_{-\infty}^{+\infty} p_t(y) \log p_t(y) dy - \int_{-\infty}^{+\infty} p_t(y) \log p_{\mathcal{M}}(y) dy. \tag{1.4}$$

Se vengono confrontati due modelli, il primo termine della (1.4) resta costante e il confronto si riduce al secondo termine della (1.4), ovvero

$$\int_{-\infty}^{+\infty} p_t(y) \log p_{\mathcal{M}}(y) dy. \tag{1.5}$$

Riscriviamo ora la (1.5) facendo riferimento alla distribuzione predittiva a posteriori, $p(\tilde{y} \mid y)$, perché ciò a cui siamo interessati è la divergenza di $p(\tilde{y} \mid y)$ da $p_t(y)$:

$$\text{elpd} = \int_{\tilde{y}} p_t(\tilde{y}) \log p(\tilde{y} \mid y) d\tilde{y}. \tag{1.6}$$

La (1.6) è chiamata *expected log predictive density* (elpd) e fornisce la risposta al problema che ci eravamo posti all'inizio di questo Capitolo, ovvero il problema di definire un criterio per valutare la capacità predittiva di un modello. Possiamo pensare alla (1.6) dicendo che descrive la distribuzione predittiva a posteriori del modello ponderando la verosimiglianza dei possibili dati futuri con la vera distribuzione p_t: all'aumentare di elpd aumenta la capacità predittiva del modello.

Non dobbiamo preoccuparci di trovare una formulazione analitica della $p(\tilde{y} \mid y)$ perché, come abbiamo visto nel Capitolo **??**, è possibile approssimare tale

distribuzione mediante simulazione. Notiamo però che la (1.6) è formulata nei termini del vero modello generatore dei dati, p_t, il quale, ovviamente, è ignoto.[2] Di conseguenza, la quantità elpd non può mai essere calcolata in maniera esatta, ma può essere solo stimata. Il secondo problema di questo Capitolo è capire come la (1.6) possa essere stimata utilizzando un campione di osservazioni.

Log pointwise predictive density

Ingenuamente, potremmo pensare di stimare la (1.6) ipotizzando che la distribuzione del campione coincida con p_t. Usare la distribuzione del campione come proxy del vero modello generatore dei dati (ovvero, ipotizzare che la distribuzione del campione rappresenti fedelmente p_t) comporta due conseguenze:

- dato che il campione è finito, anziché eseguire un'operazione di integrazione, possiamo semplicemente sommare la densità predittiva a posteriori delle osservazioni;
- non è necessario ponderare per p_t, in quanto assumiamo che la distribuzione empirica del campione corrisponda a p_t (ciò significa assumere che i valori più comunemente osservati nel campione siano anche quelli più verosimili nella vera distribuzione p_t).

Questo conduce alla seguente equazione:

$$\frac{1}{n}\sum_{i=1}^{n}\log p(y_i^{rep} \mid y). \tag{1.7}$$

La quantità (1.7), senza il passaggio finale della divisione per il numero di osservazioni, è chiamata *log pointwise predictive density* (lppd)

$$\text{lppd} = \sum_{i=1}^{n}\log p(y_i^{rep} \mid y) \tag{1.8}$$

e corrisponde alla somma delle densità predittive logaritmiche delle n osservazioni. Valori più grandi della (1.8) sono da preferire perché indicano una maggiore accuratezza media. È anche comune vedere espressa la quantità precedente nei termini della *devianza*, ovvero alla lppd moltiplicata per -2. In questo secondo caso sono da preferire valori piccoli.

È importante notare che lppd è una *sovrastima* di (1.6). Tale sovrastima è dovuta al fatto che, nel calcolo della (1.8), abbiamo usato $p(y_i^{rep} \mid y)$ al posto di $p(\tilde{y} \mid y)$: in altri termini, abbiamo considerato le osservazioni del campione come se fossero un nuovo campione di dati.

In una serie di simulazioni, McElreath (2020) esamina il significato di questa sovrastima. Nelle simulazioni la devianza viene calcolata come funzione della complessità (ovvero, il numero di parametri) del modello. La simulazione mostra che lppd aumenta al crescere della complessità del modello. Ciò significa che lppd mostra lo stesso limite del coefficiente di determinazione: aumenta all'aumentare della complessità.

[2] Se il modello sottostante i dati fosse noto non avremmo bisogno di cercare il modello migliore, perché p_t è il modello migliore.

Esempio 1.8. Esaminiamo un esempio tratto da Bayesian Data Analysis for Cognitive Science nel quale la elpd viene calcolata in forma esatta oppure mediante approssimazione.

Supponiamo di disporre di un campione di n osservazioni. Supponiamo inoltre di conoscere il vero processo generativo dei dati (qualcosa che in pratica non è mai possibile), ovvero:

$$p_t(y) = \mathrm{B}(1, 3).$$

I dati sono

```
n <- 10000
y_data <- rbeta(n, 1, 3)
head(y_data)
#> [1] 0.4540 0.1569 0.1065 0.1482 0.2831 0.0228
```

Supponiamo inoltre di avere adattato ai dati un modello bayesiano $\mathcal{M}$ e di avere ottenuto la distribuzione a posteriori per i parametri del modello. Inoltre, supponiamo di avere derivato la forma analitica della distribuzione predittiva a posteriori per il modello:

$$p(y^{rep} \mid y) \sim \mathrm{B}(2, 2).$$

Questa distribuzione ci dice quanto sono credibili i possibili dati futuri.

Conoscendo la vera distribuzione dei dati $p_t(y)$ possiamo calcolare in forma esatta la quantità elpd, ovvero

$$\text{elpd} = \int_{y^{rep}} p_t(y^{rep}) \log p(y^{rep} \mid y) dy^{rep}.$$

Svolgiamo i calcoli in R otteniamo:

```
# True distribution
p_t <- function(y) dbeta(y, 1, 3)
# Predictive distribution
p <- function(y) dbeta(y, 2, 2)
# Integration
integrand <- function(y) p_t(y) * log(p(y))
integrate(f = integrand, lower = 0, upper = 1)
#> -0.375 with absolute error < 6.8e-07
```

Tuttavia, in pratica non conosciamo mai $p_t(y)$. Quindi approssimiamo elpd usando la (1.6):

$$\frac{1}{n} \sum_{i=1}^{n} \log p(y_i \mid y).$$

Così facendo, e svolgendo i calcoli in R, otteniamo

```
1 / n * sum(log(p(y_data)))
#> [1] -0.375
```

un valore maggiore di quello trovato in precedenza.

1.8 Criterio di informazione e convalida incrociata K-fold

Nel Paragrafo precedente abbiamo visto che la (1.8) fornisce una sovrastima di elpd. Dato che, in pratica, un nuovo campione di dati futuri non è mai disponibile, sono stati messi a punto due metodi per evitare la sovrastima della elpd utilizzando soltanto le informazioni fornite dal campione osservato. Ciò si ottiene mediante:

- l'introduzione di un fattore di correzione;
- la convalida incrociata cosiddetta K-fold.

AIC, DIC e WAIC

Allo scopo di evitare la sovrastima della (1.8), le statistiche *Akaike Information Criterion* (AIC), *Deviance Information Criterion* (DIC) e *Widely Applicable Information Criterion* (WAIC) introducono un fattore di correzione. Le statistiche DIC e WAIC sono più complesse di AIC, ma producono un'approssimazione migliore. Tuttavia, i valori AIC, DIC e WAIC sono spesso molto simili tra loro. Per convenienza, ci accontenteremo qui di esaminare da vicino la statistica più semplice, ovvero AIC.

Criterio d'informazione di Akaike

Il criterio d'informazione di Akaike (in inglese *Akaike information criterion*, indicato come AIC) fornisce un metodo molto semplice per stimare la devianza media *out-of-sample*.

Definizione 1.3. Il criterio d'informazione di Akaike è definito come

$$AIC = -2 \log p(y \mid \hat{\theta}_{MLE}) + 2k, \tag{1.9}$$

dove k è il numero di parametri stimati nel modello e $p(y \mid \hat{\theta}_{MLE})$ è il valore massimizzato della funzione di verosimiglianza del modello stimato.

Dividendo per -2, otteniamo elpd_{AIC}:

$$\widehat{\text{elpd}}_{AIC} = \log p(y \mid \hat{\theta}_{MLE}) - k, \tag{1.10}$$

dove k è il fattore di correzione introdotto per evitare la sovrastima discussa in precedenza.

AIC è di interesse principalmente storico e produce una approssimazione attendibile di elpd quando:

1. le distribuzioni a priori sono non informative;
2. la distribuzione a posteriori è approssimativamente gaussiana multivariata;
3. la dimensione n del campione è molto maggiore del numero k dei parametri.

Esempio 1.9. Per meglio comprendere la statistica $\widehat{\text{elpd}}_{AIC}$, esaminiamo un esempio discusso da Gelman et al. (2014). Sia $y_1, \dots, y_n \sim \mathcal{N}(\theta, 1)$ un campione

di osservazioni. Nel caso di una distribuzione a priori non-informativa $p(\theta) \propto 1$, la stima di massima verosimiglianza è $\bar{y}$. La log-verosimiglianza è

$$\begin{aligned}\log p(y \mid \hat{\theta}_{MLE}) &= -\frac{n}{2}\log(2\pi) - \frac{1}{2}\sum_{i=1}^{n}(y_i - \bar{y})^2 \\ &= -\frac{n}{2}\log(2\pi) - \frac{1}{2}(n-1)s_y^2, \end{aligned} \tag{1.11}$$

dove s_y^2 è la varianza campionaria.

Nel caso di un modello Normale con con varianza nota e una distribuzione a priori uniforme viene stimato un solo parametro, per cui

$$\begin{aligned}\widehat{\text{elpd}}_{AIC} &= \log p(y \mid \hat{\theta}_{MLE}) - k \\ &= -\frac{n}{2}\log(2\pi) - \frac{1}{2}(n-1)s_y^2 - 1. \end{aligned} \tag{1.12}$$

Convalida incrociata K-fold

La sovrastima della (1.8) può anche essere evitata usando una tecnica chiamata *K-fold cross-validation*. Mediante questo metodo vengono stimati i parametri del modello tralasciando una porzione di osservazioni (chiamata *fold*) dal campione per poi valutare il modello sulle osservazioni che sono state escluse. Una stima complessiva dell'accuratezza si ottiene poi calcolando la media del punteggio di accuratezza ottenuto in ogni fold. Il numero minimo di fold è 2; all'altro estremo, è possibile impiegare una singola osservazione in ciascun fold e adattare il modello tante volte (n) quante sono le singole osservazioni. Questa strategia è chiamata *leave-one-out cross-validation* (LOO-CV).

Importance sampling

La strategia LOO-CV è computazionalmente onerosa (ovvero, richiede un tempo di esecuzione molto lungo). È però possibile approssimare LOO-CV mediante un metodo chiamato *Pareto-smoothed importance sampling cross-validation* [PSIS; Vehtari et al. (2017)]. Tralasciando i dettagli matematici, l'intuizione di base è che PSIS fa leva sul punteggio di "importanza" posseduto da ciascuna osservazione all'interno della distribuzione a posteriori. Per "importanza" si intende il fatto che alcune osservazioni hanno un impatto maggiore sulle proprietà della distribuzione a posteriori di altre: se viene rimossa un'osservazione importante, le proprietà della distribuzione a posteriori cambiano molto; se viene rimossa un'osservazione poco importante, la distribuzione a posteriori cambia poco. L'"importanza" così intesa viene chiamata "peso" (*weight*) e tali pesi vengono utilizzati per stimare l'accuratezza *out-of-sample* del modello. PSIS-LOO-CV richiede che il modello venga adattato una volta soltanto ai dati e fornisce una stima della devianza *out-of-sample* che evita la sovrastima della (1.8). Inoltre, PSIS-LOO-CV fornisce un feedback sulla propria affidabilità identificando le osservazioni i cui pesi molto elevati potrebbero rendere imprecisa la predizione.

Valori $\widehat{\text{elpd}}_{\text{LOO}}$ più grandi indicano una maggiore accuratezza predittiva. In alternativa, anziché considerare $\widehat{\text{elpd}}$, è possibile usare la quantità $-2 \cdot \widehat{\text{elpd}}$, la

quale è chiamata *LOO Information Criterion* (LOOIC). In questo secondo caso, valori LOOIC più piccoli sono da preferire.

La quantità $\widehat{\text{elpd}}_{\text{LOO}}$ viene calcolata dai pacchetti `loo` e `brms` ed è chiamata `elpd_loo` o `elpd_kfold`. È anche possibile calcolare la differenza della quantità `elpd_loo` per modelli alternativi, insieme alla deviazione standard della distribuzione campionaria di tale differenza.

Confronto tra AIC e LOO-CV

Per fare un esempio, faremo qui un confronto tra $\widehat{\text{elpd}}_{AIC}$ e $\widehat{\text{elpd}}_{LOO-CV}$. Esaminiamo nuovamente l'associazione tra il QI dei figli e il QI delle madri nel campione di dati discusso da Gelman et al. (2020). Una tale relazione può essere descritta da un modello di regressione nel quale la y corrisponde al QI dei figli e la x al QI delle madri.

Leggiamo i dati in R:

```
library("foreign")
df <- read.dta(here("data", "kidiq.dta"))
df$y <- scale(df$kid_score)[, 1]
df$x1 <- scale(df$mom_iq)[, 1]
head(df)
#>   kid_score mom_hs mom_iq mom_work mom_age       y      x1
#> 1        65      1  121.1        4      27 -1.0679  1.4078
#> 2        98      1   89.4        4      25  0.5489 -0.7092
#> 3        85      1  115.4        4      27 -0.0881  1.0295
#> 4        83      1   99.4        3      25 -0.1860 -0.0367
#> 5       115      1   92.7        4      27  1.3818 -0.4836
#> 6        98      0  107.9        1      18  0.5489  0.5268
```

Dato che AIC non è una statistica bayesiana, può essere calcolata mediante strumenti frequentisti:

```
m1_freq <- lm(y ~ x1, data = df)
AIC(m1_freq) / -2
#> [1] -570
```

Per ottenere LOO-CV adattiamo ai dati un modello di regressione bayesiano:

```
modelString <- "
data {
  int<lower=0> N;
  vector[N] x1;
  vector[N] y;
}
parameters {
  real alpha;
  real beta1;
  real<lower=0> sigma;
```

```
}
transformed parameters {
  vector[N] mu;
  for (n in 1:N){
    mu[n] = alpha + beta1*x1[n];
  }
}
model {
  alpha ~ normal(0, 1);
  beta1 ~ normal(0, 1);
  sigma ~ cauchy(0, 1);
  y ~ normal(mu, sigma);
}
generated quantities {
  vector[N] y_rep;
  vector[N] log_lik;
  for (n in 1:N){
    y_rep[n] = normal_rng(mu[n], sigma);
    log_lik[n] = normal_lpdf(y[n] | x1[n] * beta1, sigma);
  }
}
"
writeLines(modelString, con = "code/simplereg.stan")
```

```
data1_list <- list(
  N = length(df$kid_score),
  y = df$y,
  x1 = df$x1
)
```

```
file1 <- file.path("code", "simplereg.stan")
```

```
mod1 <- cmdstan_model(file1)
```

Eseguiamo il campionamento MCMC:

```
fit1 <- mod1$sample(
  data = data1_list,
  iter_sampling = 4000L,
  iter_warmup = 2000L,
  seed = SEED,
  chains = 4L,
  parallel_chains = 2L,
  refresh = 0,
  thin = 1
)
```

Calcoliamo infine la quantità $\widehat{\text{elpd}}_{LOO-CV}$:

```
loo1_result <- fit1$loo(cores = 4)
print(loo1_result)
#>
#> Computed from 16000 by 434 log-likelihood matrix
#>
#>          Estimate   SE
#> elpd_loo   -568.6 14.5
#> p_loo         1.9  0.2
#> looic      1137.2 28.9
#> ------
#> Monte Carlo SE of elpd_loo is 0.0.
#>
#> All Pareto k estimates are good (k < 0.5).
#> See help('pareto-k-diagnostic') for details.
```

Si noti la somiglianza tra $\widehat{\text{elpd}}_{LOO-CV}$ e $\widehat{\text{elpd}}_{AIC}$. In conclusione, possiamo dunque dire che $\widehat{\text{elpd}}_{LOO-CV}$ è la risposta bayesiana allo stesso problema che trova una soluzione frequentista nella statistica $\widehat{\text{elpd}}_{AIC}$.

Confronto tra modelli mediante LOO-CV

Come menzionato in precedenza, l'obiettivo centrale della misurazione dell'accuratezza predittiva è il confronto di modelli. Una volta capito come calcolare LOO-CV con un condice scritto in linguaggio Stan, svolgeremo ora un confronto di modelli.[3]

Considereremo qui un confronto di modelli di regressione. Il modello di regressione discusso nel Paragrafo precedente prevede il QI dei bambini dal QI delle madri. Aggiungiamo a tale modello un secondo predittore che corrisponde all'età della madre. L'aggiunta di tale predittore migliori l'accuratezza predittiva del modello?

```
modelString <- "
data {
  int<lower=0> N;
  vector[N] x1;
  vector[N] x2;
  vector[N] y;
}
parameters {
  real alpha;
  real beta1;
```

[3] A questo proposito, è necessario aggiungere una nota di cautela. Come fa notare McElreath (2020), fare previsioni e inferire i rapporti causali sono due cose molto diverse. Statistiche quali AIC, WAIC e LOO-CV consentono di individuare modelli con buone capacità predittive. Tali modelli, tuttavia, non riflettono necessariamente la struttura causale del fenomeno considerato: la selezione di modelli basata unicamente sull'accuratezza predittiva non garantisce che venga selezionato il modello che riflette la struttura causale del fenomeno (si veda anche Navarro, 2019).

```
  real beta2;
  real<lower=0> sigma;
}
transformed parameters {
  vector[N] mu;
  for (n in 1:N){
    mu[n] = alpha + beta1*x1[n] + beta2*x2[n];
  }
}
model {
  alpha ~ normal(0, 1);
  beta1 ~ normal(0, 1);
  beta2 ~ normal(0, 1);
  sigma ~ cauchy(0, 1);
  y ~ normal(mu, sigma);
}
generated quantities {
  vector[N] y_rep;
  vector[N] log_lik;
  for (n in 1:N){
    y_rep[n] = normal_rng(mu[n], sigma);
    log_lik[n] = normal_lpdf(y[n] | x1[n] * beta1 + x2[n] * beta2, sigma);
  }
}
"
writeLines(modelString, con = "code/mreg2.stan")
```

```
df$x2 <- scale(df$mom_age)[, 1]
```

```
data2_list <- list(
  N = length(df$kid_score),
  y = df$y,
  x1 = df$x1,
  x2 = df$x2
)
```

```
file2 <- file.path("code", "mreg2.stan")
```

```
# compile model
mod2 <- cmdstan_model(file2)
```

```
# Running MCMC
fit2 <- mod2$sample(
  data = data2_list,
  iter_sampling = 4000L,
  iter_warmup = 2000L,
  seed = SEED,
```

```
  chains = 4L,
  parallel_chains = 2L,
  refresh = 0,
  thin = 1
)
```

```
fit2$summary(c("alpha", "beta1", "beta2", "sigma"))
#> # A tibble: 4 x 10
#>   variable     mean  median     sd    mad      q5    q95  rhat ess_bulk ess_tail
#>   <chr>       <dbl>   <dbl>  <dbl>  <dbl>   <dbl>  <dbl> <dbl>    <dbl>    <dbl>
#> 1 alpha    0.000387 5.70e-4 0.0431 0.0427 -0.0706 0.0709  1.00   18092.   12482.
#> 2 beta1    0.442    4.42e-1 0.0434 0.0428  0.372  0.514   1.00   18884.   12262.
#> 3 beta2    0.0510   5.11e-2 0.0431 0.0431 -0.0192 0.122   1.00   19099.   12929.
#> 4 sigma    0.896    8.96e-1 0.0306 0.0303  0.847  0.947   1.00   18776.   13031.
```

```
loo2_result <- fit2$loo(cores = 4)
print(loo2_result)
#>
#> Computed from 16000 by 434 log-likelihood matrix
#>
#>          Estimate   SE
#> elpd_loo   -569.0 14.5
#> p_loo         3.0  0.3
#> looic      1137.9 29.0
#> ------
#> Monte Carlo SE of elpd_loo is 0.0.
#>
#> All Pareto k estimates are good (k < 0.5).
#> See help('pareto-k-diagnostic') for details.
```

Consideriamo infine un terzo modello che utilizza come predittori, oltre al QI della madre, una variabile dicotomica (codificata 0 o 1) che distingue madri che hanno completato le scuole superiori da quelle che non le hanno completate. Nuovamente, la domanda è se l'aggiunta di tale predittore migliori la capacità predittiva del modello.

```
modelString <- "
data {
  int<lower=0> N;
  vector[N] x1;
  vector[N] x3;
  vector[N] y;
}
parameters {
  real alpha;
  real beta1;
  real beta3;
  real<lower=0> sigma;
```

```
}
transformed parameters {
  vector[N] mu;
  for (n in 1:N){
    mu[n] = alpha + beta1*x1[n] + beta3*x3[n];
  }
}
model {
  alpha ~ normal(0, 1);
  beta1 ~ normal(0, 1);
  beta3 ~ normal(0, 1);
  sigma ~ cauchy(0, 1);
  y ~ normal(mu, sigma);
}
generated quantities {
  vector[N] y_rep;
  vector[N] log_lik;
  for (n in 1:N){
    y_rep[n] = normal_rng(mu[n], sigma);
    log_lik[n] = normal_lpdf(y[n] | x1[n] * beta1 + x3[n] * beta3, sigma);
  }
}
"
writeLines(modelString, con = "code/mreg3.stan")
```

```
df$x3 <- df$mom_hs
```

```
data3_list <- list(
  N = length(df$kid_score),
  y = df$y,
  x1 = df$x1,
  x3 = df$x3
)
```

```
file3 <- file.path("code", "mreg3.stan")
```

```
mod3 <- cmdstan_model(file3)
```

```
fit3 <- mod3$sample(
  data = data3_list,
  iter_sampling = 4000L,
  iter_warmup = 2000L,
  seed = SEED,
  chains = 4L,
  parallel_chains = 2L,
  refresh = 0,
  thin = 1
)
```

```
fit3$summary(c("alpha", "beta1", "beta3", "sigma"))
#> # A tibble: 4 x 10
#>   variable   mean median     sd    mad     q5     q95  rhat ess_bulk ess_tail
#>   <chr>     <dbl>  <dbl>  <dbl>  <dbl>  <dbl>   <dbl> <dbl>    <dbl>    <dbl>
#> 1 alpha    -0.225 -0.225 0.0951 0.0939 -0.380 -0.0673  1.00    7808.    8235.
#> 2 beta1     0.414  0.414 0.0445 0.0440  0.340  0.487   1.00   10200.    9870.
#> 3 beta3     0.287  0.288 0.108  0.106   0.108  0.463   1.00    7832.    8542.
#> 4 sigma     0.890  0.889 0.0300 0.0295  0.842  0.941   1.00   11733.   10064.
```

```
loo3_result <- fit3$loo(cores = 4)
print(loo3_result)
#>
#> Computed from 16000 by 434 log-likelihood matrix
#>
#>          Estimate   SE
#> elpd_loo   -584.2 16.4
#> p_loo         7.4  0.6
#> looic      1168.4 32.8
#> ------
#> Monte Carlo SE of elpd_loo is 0.0.
#>
#> All Pareto k estimates are good (k < 0.5).
#> See help('pareto-k-diagnostic') for details.
```

Per eseguire un confronto tra modelli in termini della loro capacità predittiva esaminiamo la differenza di LOO-CV tra coppie di modelli. Le seguenti istruzioni R producono la quantità `elpd_diff`, ovvero la differenza tra stime della elpd fornite da due modelli. Il primo argomento della funzione `loo_compare()` specifica il modello che viene usato come confronto. Nella prima riga dell'output, il valore `elpd_diff` è 0 (cioè, $x - x = 0$). Nelle righe successive sono riportate le differenze rispetto al modello di confronto (in questo caso, il modello 1). La colonna `se_diff` riporta l'errore standard di tali differenze.

L'incertezza della stima dell'accuratezza *out-of-sample* si distribuisce in maniera approssimativamente normale con media uguale al valore riportato dal software e deviazione standard uguale a ciò che è indicato nell'output come errore standard. Quando il campione è piccolo, questa approssimazione produce una forte sottostima dell'incertezza, ma fornisce comunque una stima migliore di AIC, DIC e WAIC.

```
w <- loo_compare(loo1_result, loo2_result, loo3_result)
print(w)
#>        elpd_diff se_diff
#> model1   0.0       0.0
#> model2  -0.4       1.3
#> model3 -15.6       6.0
```

Per interpretare l'output, usiamo il criterio suggerito da Gelman et al. (1995): consideriamo "credibile" una differenza se `elpd_diff` è almeno due volte maggiore

di `se_diff`. Nel caso presente, dunque, il confronto tra il modello 2 e il modello 1 indica che la quantità `elpd_diff` è molto piccola rispetto al suo errore standard. Questo accade se un predittore è associato in modo trascurabile con la variabile dipendente. I dati presenti, dunque, non offrono alcuna evidenza che aggiungere dell'età della madre come predittore migliori la capacità predittiva del modello. Nel confronto tra modello 3 e modello 1, invece, la quantità `elpd_diff` è maggiore di due volte il valore dell'errore standard. Questo suggerisce un incremento della capacità predittiva del modello quiando il livello di istruzione della madre viene incluso tra i predittori.

È anche possibile calcolare l'intervallo di credibilità per `elpd_diff`:

```
15.5 + c(-1, 1) * qnorm(.95, 0, 1) * 6.0
#> [1]  5.63 25.37
```

Outier

Si è soliti pensare che la maggior parte delle osservazioni del campione sia prodotta da un unico meccanismo generatore dei dati, mentre le rimanenti osservazioni sono la realizzazione di un diverso processo stocastico. Le osservazioni che appartengono a questo secondo gruppo si chiamano *outlier*. È dunque necessario identificare gli outlier e limitare la loro influenza sull'inferenza.[4]

Poniamoci ora il problema di identificare gli outlier con la tecnica PSIS-LOO-CV. Quando PSIS-LOO-CV viene calcolato con il pacchetto `loo`, l'output riporta il parametro di forma della distribuzione di Pareto (valore `k`). Tale valore può essere utilizzato per identificare gli outlier. Infatti, il valore `k` valuta, per ciascun punto del campione, l'approssimazione usata da PSIS-LOO-CV. Se $k < 0.5$, i pesi di importanza vengono stimati in modo accurato; se il valore k di Pareto di un punto è > 0.7, i pesi di importanza possono essere inaccurati. Le osservazioni con $k > 0.7$ sono dunque osservazioni outlier.

Per fare un esempio concreto, introduciamo nel campione dell'esempio precedente una singola osservazione outlier.

```
df1 <- df
dim(df1)
#> [1] 434   9
df1$x1[434] <- 10
df1$y[434] <- 10
```

Sistemiamo i dati nel formato appropriato per Stan:

[4] McElreath (2020) nota che, spesso, i ricercatori eliminano i valori anomali prima di adattare un modello ai dati, basandosi solo sulla distanza dal valore medio della variabile dipendente misurata in termini di unità di deviazione standard. Secondo McElreath (2020) questo non dovrebbe mai essere fatto: un'osservazione può essere considerata come un valore anomalo o un valore influente solo alla luce delle predizioni di un modello (mai prima di avere adattato il modello ai dati). Se ci sono solo pochi valori anomali una strategia possibile è quella di riportare i risultati delle analisi statistiche svolte su tutto il campione dei dati oppure dopo avere eliminato le osservazioni anomale e influenti.

```
data1a_list <- list(
  N = length(df1$kid_score),
  y = df1$y,
  x1 = df1$x1
)
```

Adattiamo nuovamente il modello 1 ad un campione di dati che contiene un outlier.

```
fit1a <- mod1$sample(
  data = data1a_list,
  iter_sampling = 4000L,
  iter_warmup = 2000L,
  seed = SEED,
  chains = 4L,
  parallel_chains = 2L,
  refresh = 0,
  thin = 1
)
```

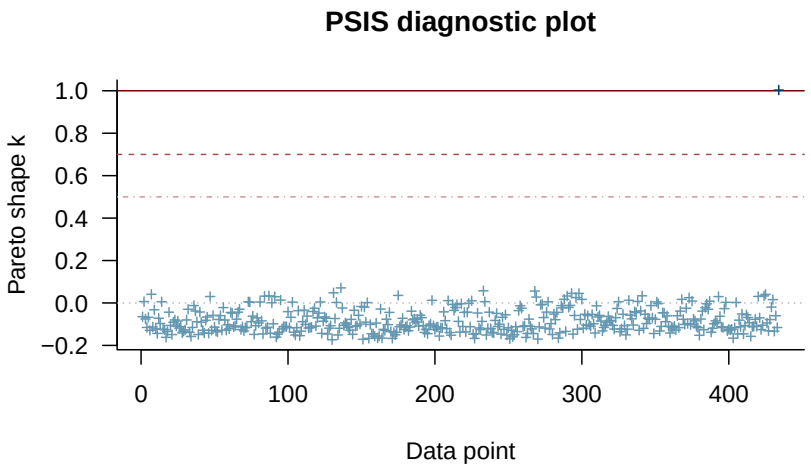
```
loo1a_result <- fit1a$loo(cores = 4)
```

Una tabella diagnostica che riassume le stime dei parametri di forma della distribuzione di Pareto si ottiene nel modo seguente:

```
print(loo1a_result)
#>
#> Computed from 16000 by 434 log-likelihood matrix
#>
#>          Estimate   SE
#> elpd_loo   -586.6 20.1
#> p_loo         7.1  5.4
#> looic      1173.2 40.3
#> ------
#> Monte Carlo SE of elpd_loo is NA.
#>
#> Pareto k diagnostic values:
#>                          Count Pct.    Min. n_eff
#> (-Inf, 0.5]   (good)     433   99.8%   9998
#>  (0.5, 0.7]   (ok)         0    0.0%   <NA>
#>    (0.7, 1]   (bad)        0    0.0%   <NA>
#>    (1, Inf)   (very bad)   1    0.2%   13
#> See help('pareto-k-diagnostic') for details.
```

Un grafico che riporta le stime dei parametri di forma della distribuzione di Pareto per ciascuna osservazione è dato da:

```
plot(loo1a_result)
```



Il valore `k` stimato da PSIS-LOO-CV mette chiaramente in luce il fatto che il valore introdotto nel campione è un outlier. L'indice dell'osservazione outlier è identificato con:

```
pareto_k_ids(loo1a_result, threshold = 0.7)
#> [1] 434
```

Considerazioni conclusive

Dati due modelli computazionali che forniscono resoconti diversi di un set di dati, come possiamo decidere quale modello è maggiormente supportato dai dati? Nel presente Capitolo abbiamo visto come il problema del confronto di modelli possa essere formulato nei termini di un problema di inferenza statistica. È però necessaria una nota di cautela. Navarro (2019) ci fa notare che il problema statistico del confronto di modelli non risolve il problema scientifico della selezione di teorie. A questo proposito usa una citazione di George Box:

> Since all models are wrong the scientist must be alert to what is importantly wrong. It is inappropriate to be concerned about mice when there are tigers abroad.

Quali sono le tigri a cui fa riferimento George Box? Corrispondono evidentemente all'assunzione che sta alla base delle procedure discusse in questo Capitolo, ovvero all'ipotesi che il vero meccanismo generatore dei dati sia noto e che l'unica incognita siano i parametri. Tuttavia le cose non sono così semplici: nei casi di interesse scientifico è lo stesso meccanismo generatore dei dati ad essere sconosciuto. I ricercatori non comprendono appieno i fenomeni che stanno studiando (altrimenti perché studiarli?) e qualunque descrizione formale di un fenomeno (modello) è sbagliata in un modo sconosciuto e sistematico. Di conseguenza, è "facile" fare inferenza sulla capacità predittiva del modello, ma è molto difficile fare inferenza sulla struttura causale dei fenomeni. In altre parole, se le analisi statistiche ci dicono che un modello ha una buona accuratezza predittiva, con ciò

non abbiamo imparato nulla sulla struttura causale del fenomeno. Ma è anche vera l'affermazione opposta: un modello che non ha *neppure* una buona accuratezza predittiva esso è sicuramente inutile: non è in grado né di fare previsioni accurate né di catturare la struttura causale.

Bibliografia

Burger, E. B. & Starbird, M. (2012). *The 5 elements of effective thinking*. Princeton University Press. (Cit. a p. ix).

Gelman, A., Carlin, J. B., Stern, H. S. & Rubin, D. B. (1995). *Bayesian data analysis*. Chapman; Hall/CRC. (Cit. a p. 21).

Gelman, A., Hill, J. & Vehtari, A. (2020). *Regression and other stories*. Cambridge University Press. (Cit. a p. 15).

Gelman, A., Hwang, J. & Vehtari, A. (2014). Understanding predictive information criteria for Bayesian models. *Statistics and Computing*, *24*(6), 997–1016 (cit. a p. 13).

Horn, S. & Loewenstein, G. (2021). Underestimating Learning by Doing. *Available at SSRN 3941441* (cit. a p. x).

Johnson, A. A., Ott, M. & Dogucu, M. (2022). *Bayes Rules! An Introduction to Bayesian Modeling with R*. CRC Press. (Cit. a p. 1).

McElreath, R. (2020). *Statistical rethinking: A Bayesian course with examples in R and Stan* (2nd Edition). Boca Raton, Florida, CRC Press. (Cit. alle pp. 2–4, 7, 11, 17, 22).

Navarro, D. J. (2019). Between the devil and the deep blue sea: Tensions between scientific judgement and statistical model selection. *Computational Brain & Behavior*, *2*(1), 28–34 (cit. alle pp. 17, 24).

Vehtari, A., Gelman, A. & Gabry, J. (2017). Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC. *Statistics and computing*, *27*(5), 1413–1432 (cit. a p. 14).

Elenco delle figure

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

Keywords Data science, Bayesian statistics.