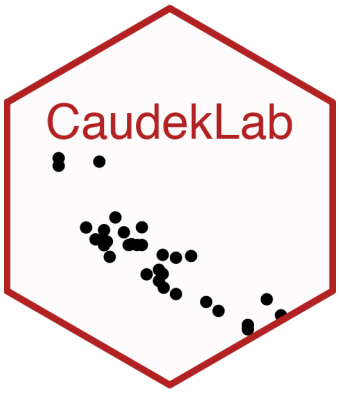
Psicometria

Corrado Caudek

2021-11-21

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 21, 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 psico-

logico 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

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

in maniera approssimativa questo punto, non devo preoccuparmi del resto". Ma in realtà non è vero: se la nostra comprensione è superficiale, quando il problema verrà presentato in una nuova forma, non riusciremo a risolverlo. Per cui i dubbi che ci vengono quando studiamo qualcosa sono il nostro alleato più prezioso: ci dicono esattamente quali sono gli aspetti che dobbiamo approfondire per potere migliorare la nostra preparazione.

- È utile sviluppare una visione d'insieme degli argomenti trattati, capire l'obiettivo generale che si vuole raggiungere e avere chiaro il contributo che i vari pezzi di informazione forniscono al raggiungimento di tale obiettivo. Questa organizzazione mentale del materiale di studio facilita la comprensione. È estremamente utile creare degli schemi di ciò che si sta studiando. Non aspettate che sia io a fornirvi un riepilogo di ciò che dovete imparare: sviluppate da soli tali schemi e tali riassunti.
- Tutti noi dobbiamo imparare l'arte di trovare le informazioni, non solo nel caso di questo insegnamento. Quando vi trovate di fronte a qualcosa che non capite, o ottenete un oscuro messaggio di errore da un software, ricordatevi: "Google is your friend".

Corrado Caudek

Febbraio 2022

CAPITOLO 1

Adattare il modello di regressione ai dati

In questo capitolo verranno esposte alcune nozioni matematiche che stanno alla base dell'inferenza per i modelli di regressione e un po' di algebra che ci aiuterà a comprendere la stima della regressione lineare. Spiegheremo anche la logica per l'uso della funzione bayesiana [`brm()`] e la sua connessione con la regressione lineare classica. Questo capitolo fornisce quindi lo sfondo e la motivazione per la discussione dei capitoli successivi.

1.1 Minimi quadrati

Nel modello di regressione lineare classico, $y_i = a + bx_i + \epsilon_i$, i coefficienti a e b sono stimati in modo da minimizzare gli errori ϵ_i. Se il numero dei dati n è maggiore di 2, non è generalmente possibile trovare una retta che passi per tutte le osservazioni (x, y) (sarebbe $y_i = a + bx_i$, senza errori, per tutti i punti $i = 1, \dots, n$), e l'obiettivo della stima è scegliere i valori $(\hat{a}, \hat{b})$ che minimizzano la somma dei quadrati dei residui,

$$r_i = y_i - (\hat{a} + \hat{b}x_i).$$

Distinguiamo tra i residui $r_i = y_i - (\hat{a} + \hat{b}x_i)$ e gli *errori* $\epsilon_i = y_i - (a + bx_i)$. Il modello è scritto in termini degli errori, ma possiamo solo lavorare con i residui: non possiamo calcolare gli errori perché per farlo sarebbe necessario conoscere a e b.

La somma dei residui quadratici (*residual sum of squares*) è

$$\text{RSS} = \sum_{i=1}^{n} (y_i = (\hat{a} + \hat{b}x_i))^2.$$

I coefficienti $(\hat{a}, \hat{b})$ che minimizzano RSS sono chiamati stime dei minimi quadrati, o minimi quadrati ordinari (*ordinari least squares*), o stime OLS. ### Stima della deviazione standard dei residui σ

Nel modello di regressione, gli errori ϵ_i provengono da una distribuzione con media 0 e deviazione standard σ: la media è zero per definizione (qualsiasi media diversa da zero viene assorbita nell'intercetta, a), e la deviazione standard degli errori può essere stimata dai dati. Un modo naturale per stimare σ sarebbe semplicemente prendere la deviazione standard dei residui, $\sqrt{\frac{1}{n}\sum_{i=1}^{n} r_i^2} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} y_i - (\hat{a} + \hat{b}x_i))^2}$, ma questo sottostimerebbe leggermente σ a causa del *overfitting*, in quanto i coefficienti $\hat{a}$ e $\hat{b}$ sono stati stimati dai dati per minimizzare la somma dei residui quadratici. La correzione

standard per questo overfitting consiste nel sostituire n con $n-2$ al denominatore (con la sottrazione di 2 derivante dalla stima di due coefficienti nel modello, l'intercetta e la pendenza). Così otteniamo

$$\hat{\sigma} = \sqrt{\frac{1}{n-2}\sum_{i=1}^{n}(y_i - (\hat{a} + \hat{b}x_i))^2}.$$

Quando $n = 1$ o 2 questa espressione è priva di significato, il che ha senso: con solo due osservazioni è possibile adattare esattamente una retta al diagramma di dispersione e quindi non c'è modo di stimare l'errore dai dati.

1.2 Calcolare la somma dei quadrati

Seguendo Solomon Kurz, creiamo una funzione per calcolare la somma dei quadrati per diversi valori di a e b:

```
rss <- function(x, y, a, b) {
  # x and y are vectors,
  # a and b are scalars
  resid <- y - (a + b * x)
  return(sum(resid^2))
  }
```

Useremo i dati ….

```
df <- read.dta(here("data", "kidiq.dta"))
head(df)
#>   kid_score mom_hs mom_iq mom_work mom_age
#> 1        65      1  121.1        4      27
#> 2        98      1   89.4        4      25
#> 3        85      1  115.4        4      27
#> 4        83      1   99.4        3      25
#> 5       115      1   92.7        4      27
#> 6        98      0  107.9        1      18
```

Nell'esempio, `kid_score` è la variabile y e `mom_iq` è il predittore. Le stime dei minimi quadrati sono fornite dalla funzione `lm()`:

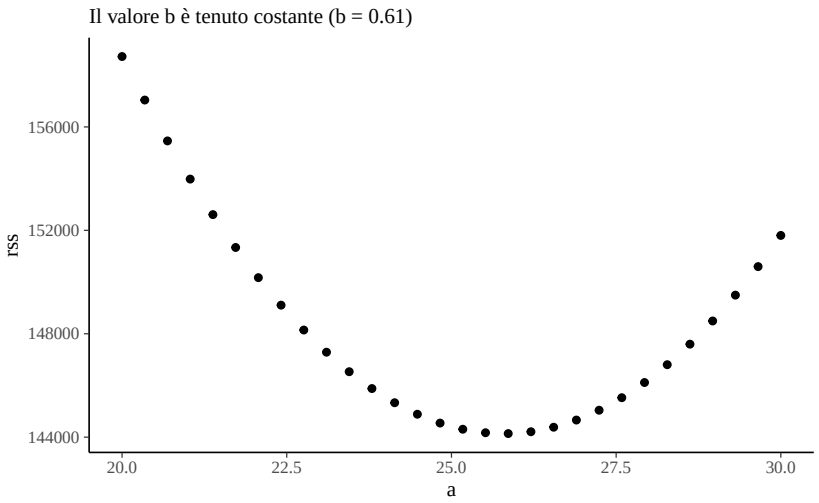
```
fm <- lm(kid_score ~ mom_iq, data = df)
coef(fm)
#> (Intercept)      mom_iq
#>       25.80        0.61
```

Calcoliamo la somma dei residui quadratici in base al modello di regressione $y_i = 25.8 + 0.61x_i + \epsilon_i$.

```
rss(df$mom_iq, df$kid_score, 25.8, 0.61)
#> [1] 144137
```

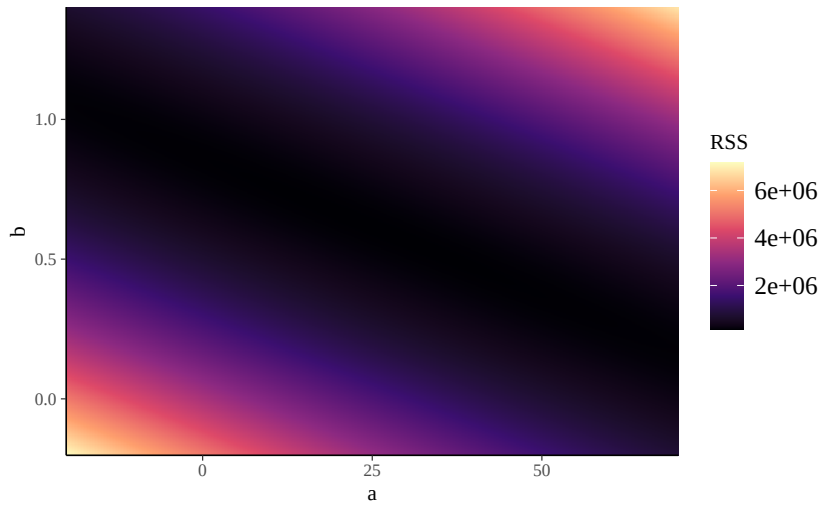
Esploriamo ora i valori assunti da rss per diversi valori di a e b. Per iniziare, utilizziamo un vettore di valori `a`, mantenendo costante `b = 0.61`.

```r
# set the global plotting theme
# theme_set(theme_linedraw() +
#             theme(panel.grid = element_blank()))
# simulate
tibble(a = seq(20, 30, length.out = 30)) %>%
  mutate(
    rss = map_dbl(
      a,
      rss,
      x = df$mom_iq,
      y = df$kid_score,
      b = 0.61)
    ) %>%

  # plot
  ggplot(aes(x = a, y = rss)) +
  geom_point() +
  labs(subtitle = "Il valore b è tenuto costante (b = 0.61)")
```



Ora variamo sia `a` che `b`, facendo assumere a ciascun parametro un insieme di valori in un intevallo, e rappresentiamo i risultati in una heat map che rappresenta l'intensitè di `rss` in funzione dei valori `a` e `b`.

```r
d <-
  crossing(a = seq(-20, 70, length.out = 400),
           b = seq(-0.2, 1.4, length.out = 400)) %>%
  mutate(rss = map2_dbl(a, b, rss, x = df$mom_iq, y = df$kid_score))
d %>%
  ggplot(aes(x = a, y = b, fill = rss)) +
  geom_tile() +
  scale_fill_viridis_c("RSS", option = "A") +
  scale_x_continuous(expand = c(0, 0)) +
  scale_y_continuous(expand = c(0, 0))
```



Poiché la stima dei minimi quadrati enfatizza il valore RSS più piccolo, la soluzione che cerchiamo corrisponde alle combinazione di `a` e `b` nell'intervallo più scuro rappresentato nella figura.

Tra gli `a` e `b` che abbiamo preso in considerazione, la coppia di valori dei parametri a cui è associato il valore `rss` più basso si trova nel modo seguente:

```
d %>%
  arrange(rss) %>%
  slice(1)
#> # A tibble: 1 x 3
#>       a     b     rss
#>   <dbl> <dbl>   <dbl>
#> 1  25.8 0.610 144137.
```

Si noti la corrispondenza tra la soluzione qui trovata e l'output della funzione `lm()`.

1.3 Massima verosimiglianza

Se gli errori del modello lineare sono indipendenti e distribuiti normalmente, in modo che $y_i \sim \mathcal{N}(a + bx_i, \sigma^2)$ per ogni i, allora la stima ai minimi quadrati di (`a`, `b`) è anche la stima di massima verosimiglianza. La *funzione di verosimiglianza* in un modello di regressione è definita come la densità di probabilità delle osservazioni dati i parametri e i predittori; quindi, in questo esempio,

$$p(y \mid a, b, \sigma, x) = \prod_{i=1}^{n} \mathcal{N}(y_i \mid a + bx_i, \sigma^2),$$

dove $\mathcal{N}(\cdot|\cdot, \cdot)$ è la funzione di densità di probabilità normale,

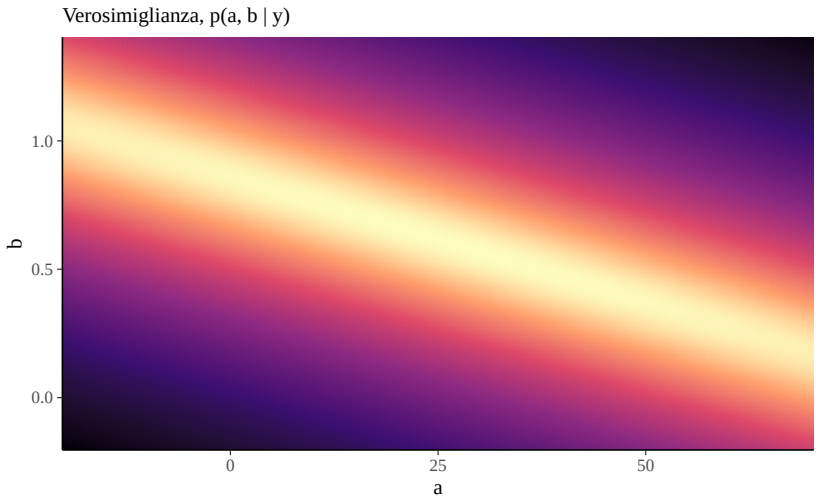
$$\mathcal{N}(y \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma}} \exp\left(-\frac{1}{2}\left(\frac{y-\mu}{\sigma}\right)^2\right).$$

Un studio della funzione precedente rivela che la massimizzazione della verosimiglianza richiede la minimizzazione della somma dei quadrati dei residui; quindi la stima dei minimi quadrati $\hat{\beta} = (\hat{a}, \hat{b})$ può essere vista come una stima di massima verosimiglianza nel modello normale:

```
ll <- function(x, y, a, b) {
  resid <- y - (a + b * x)
  sigma <- sqrt(sum(resid^2) / length(x))
  d <- dnorm(y, mean = a + b * x, sd = sigma, log = TRUE)
  tibble(sigma = sigma, ll = sum(d))
}
```

Calcoliamo dunque le stime di verosimiglianza logaritmica per varie combinazioni di (`a`, `b`), date due colonne di dati, `x` e `y`. La funzione resituisce anche il valore $\hat{\sigma}$.

```
d <-
  crossing(a = seq(-20, 70, length.out = 200),
           b = seq(-0.2, 1.4, length.out = 200)) %>%
  mutate(ll = map2(a, b, ll, x = df$mom_iq, y = df$kid_score)) %>%
  unnest(ll)
p1 <-
  d %>%
  ggplot(aes(x = a, y = b, fill = ll)) +
  geom_tile() +
  scale_fill_viridis_c(option = "A", breaks = NULL) +
  scale_x_continuous(expand = c(0, 0)) +
  scale_y_continuous(expand = c(0, 0)) +
  labs(subtitle = "Verosimiglianza, p(a, b | y)")
p1
```



Le stime di $\hat{a}, \hat{b}$ ottenute mediante il metodo di massima verosimiglianza sono:

```
d %>%
  arrange(desc(ll)) %>%
  slice(1)
#> # A tibble: 1 x 4
#>       a     b sigma     ll
#>   <dbl> <dbl> <dbl>  <dbl>
#> 1  25.7 0.612  18.2 -1876.
```

Inferenza bayesiana

Usiamo ora la funzione `brms::brm()` per eseguire l'analisi mediante un approccio bayesiano:

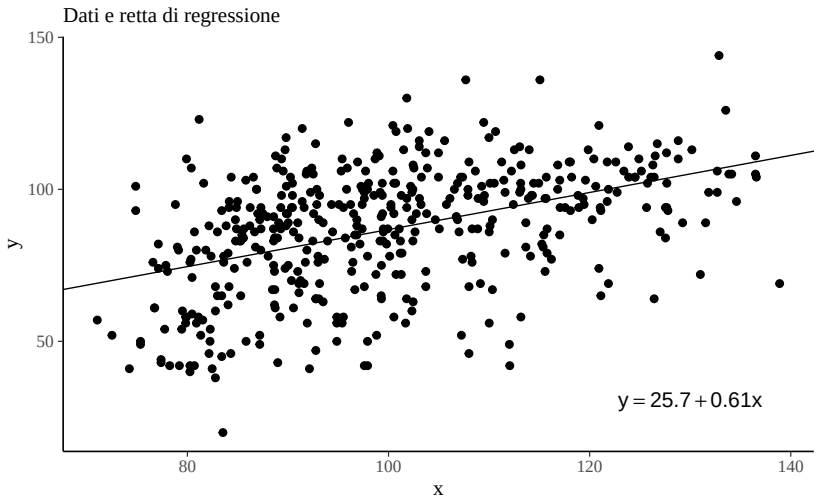
```r
m <-
  brm(
    kid_score ~ mom_iq,
    data = df,
    backend = "cmdstan"
  )
```

```
#> Running MCMC with 4 chains, at most 8 in parallel...
#>
#> Chain 1 Iteration:    1 / 2000 [  0%]  (Warmup)
#> Chain 1 Iteration:  100 / 2000 [  5%]  (Warmup)
#> Chain 1 Iteration:  200 / 2000 [ 10%]  (Warmup)
#> Chain 1 Iteration:  300 / 2000 [ 15%]  (Warmup)
#> Chain 1 Iteration:  400 / 2000 [ 20%]  (Warmup)
#> Chain 1 Iteration:  500 / 2000 [ 25%]  (Warmup)
#> Chain 1 Iteration:  600 / 2000 [ 30%]  (Warmup)
#> Chain 1 Iteration:  700 / 2000 [ 35%]  (Warmup)
#> Chain 1 Iteration:  800 / 2000 [ 40%]  (Warmup)
#> Chain 1 Iteration:  900 / 2000 [ 45%]  (Warmup)
#> Chain 1 Iteration: 1000 / 2000 [ 50%]  (Warmup)
#> Chain 1 Iteration: 1001 / 2000 [ 50%]  (Sampling)
#> Chain 1 Iteration: 1100 / 2000 [ 55%]  (Sampling)
#> Chain 1 Iteration: 1200 / 2000 [ 60%]  (Sampling)
#> Chain 1 Iteration: 1300 / 2000 [ 65%]  (Sampling)
#> Chain 1 Iteration: 1400 / 2000 [ 70%]  (Sampling)
#> Chain 1 Iteration: 1500 / 2000 [ 75%]  (Sampling)
#> Chain 1 Iteration: 1600 / 2000 [ 80%]  (Sampling)
#> Chain 1 Iteration: 1700 / 2000 [ 85%]  (Sampling)
#> Chain 1 Iteration: 1800 / 2000 [ 90%]  (Sampling)
#> Chain 1 Iteration: 1900 / 2000 [ 95%]  (Sampling)
#> Chain 1 Iteration: 2000 / 2000 [100%]  (Sampling)
#> Chain 2 Iteration:    1 / 2000 [  0%]  (Warmup)
#> Chain 2 Iteration:  100 / 2000 [  5%]  (Warmup)
#> Chain 2 Iteration:  200 / 2000 [ 10%]  (Warmup)
#> Chain 2 Iteration:  300 / 2000 [ 15%]  (Warmup)
#> Chain 2 Iteration:  400 / 2000 [ 20%]  (Warmup)
#> Chain 2 Iteration:  500 / 2000 [ 25%]  (Warmup)
#> Chain 2 Iteration:  600 / 2000 [ 30%]  (Warmup)
#> Chain 2 Iteration:  700 / 2000 [ 35%]  (Warmup)
#> Chain 2 Iteration:  800 / 2000 [ 40%]  (Warmup)
#> Chain 2 Iteration:  900 / 2000 [ 45%]  (Warmup)
#> Chain 2 Iteration: 1000 / 2000 [ 50%]  (Warmup)
#> Chain 2 Iteration: 1001 / 2000 [ 50%]  (Sampling)
#> Chain 2 Iteration: 1100 / 2000 [ 55%]  (Sampling)
#> Chain 2 Iteration: 1200 / 2000 [ 60%]  (Sampling)
#> Chain 2 Iteration: 1300 / 2000 [ 65%]  (Sampling)
#> Chain 2 Iteration: 1400 / 2000 [ 70%]  (Sampling)
#> Chain 2 Iteration: 1500 / 2000 [ 75%]  (Sampling)
#> Chain 2 Iteration: 1600 / 2000 [ 80%]  (Sampling)
#> Chain 2 Iteration: 1700 / 2000 [ 85%]  (Sampling)
#> Chain 2 Iteration: 1800 / 2000 [ 90%]  (Sampling)
#> Chain 2 Iteration: 1900 / 2000 [ 95%]  (Sampling)
#> Chain 2 Iteration: 2000 / 2000 [100%]  (Sampling)
#> Chain 3 Iteration:    1 / 2000 [  0%]  (Warmup)
#> Chain 3 Iteration:  100 / 2000 [  5%]  (Warmup)
#> Chain 3 Iteration:  200 / 2000 [ 10%]  (Warmup)
#> Chain 3 Iteration:  300 / 2000 [ 15%]  (Warmup)
#> Chain 3 Iteration:  400 / 2000 [ 20%]  (Warmup)
#> Chain 3 Iteration:  500 / 2000 [ 25%]  (Warmup)
#> Chain 3 Iteration:  600 / 2000 [ 30%]  (Warmup)
#> Chain 3 Iteration:  700 / 2000 [ 35%]  (Warmup)
#> Chain 3 Iteration:  800 / 2000 [ 40%]  (Warmup)
#> Chain 3 Iteration:  900 / 2000 [ 45%]  (Warmup)
#> Chain 3 Iteration: 1000 / 2000 [ 50%]  (Warmup)
#> Chain 3 Iteration: 1001 / 2000 [ 50%]  (Sampling)
#> Chain 3 Iteration: 1100 / 2000 [ 55%]  (Sampling)
#> Chain 3 Iteration: 1200 / 2000 [ 60%]  (Sampling)
#> Chain 3 Iteration: 1300 / 2000 [ 65%]  (Sampling)
#> Chain 3 Iteration: 1400 / 2000 [ 70%]  (Sampling)
#> Chain 3 Iteration: 1500 / 2000 [ 75%]  (Sampling)
#> Chain 3 Iteration: 1600 / 2000 [ 80%]  (Sampling)
#> Chain 3 Iteration: 1700 / 2000 [ 85%]  (Sampling)
#> Chain 3 Iteration: 1800 / 2000 [ 90%]  (Sampling)
#> Chain 3 Iteration: 1900 / 2000 [ 95%]  (Sampling)
#> Chain 3 Iteration: 2000 / 2000 [100%]  (Sampling)
#> Chain 4 Iteration:    1 / 2000 [  0%]  (Warmup)
#> Chain 4 Iteration:  100 / 2000 [  5%]  (Warmup)
#> Chain 4 Iteration:  200 / 2000 [ 10%]  (Warmup)
#> Chain 4 Iteration:  300 / 2000 [ 15%]  (Warmup)
#> Chain 4 Iteration:  400 / 2000 [ 20%]  (Warmup)
```

```
#> Chain 4 Iteration:  500 / 2000 [ 25%]  (Warmup)
#> Chain 4 Iteration:  600 / 2000 [ 30%]  (Warmup)
#> Chain 4 Iteration:  700 / 2000 [ 35%]  (Warmup)
#> Chain 4 Iteration:  800 / 2000 [ 40%]  (Warmup)
#> Chain 4 Iteration:  900 / 2000 [ 45%]  (Warmup)
#> Chain 4 Iteration: 1000 / 2000 [ 50%]  (Warmup)
#> Chain 4 Iteration: 1001 / 2000 [ 50%]  (Sampling)
#> Chain 4 Iteration: 1100 / 2000 [ 55%]  (Sampling)
#> Chain 4 Iteration: 1200 / 2000 [ 60%]  (Sampling)
#> Chain 4 Iteration: 1300 / 2000 [ 65%]  (Sampling)
#> Chain 4 Iteration: 1400 / 2000 [ 70%]  (Sampling)
#> Chain 4 Iteration: 1500 / 2000 [ 75%]  (Sampling)
#> Chain 4 Iteration: 1600 / 2000 [ 80%]  (Sampling)
#> Chain 4 Iteration: 1700 / 2000 [ 85%]  (Sampling)
#> Chain 4 Iteration: 1800 / 2000 [ 90%]  (Sampling)
#> Chain 4 Iteration: 1900 / 2000 [ 95%]  (Sampling)
#> Chain 4 Iteration: 2000 / 2000 [100%]  (Sampling)
#> Chain 1 finished in 0.0 seconds.
#> Chain 2 finished in 0.0 seconds.
#> Chain 3 finished in 0.0 seconds.
#> Chain 4 finished in 0.0 seconds.
#>
#> All 4 chains finished successfully.
#> Mean chain execution time: 0.0 seconds.
#> Total execution time: 0.3 seconds.
```

Utilizzando i coefficienti calcolati da `brms::brm()`, aggiungiamo la stima della retta di regressione al diagramma di dispersione dei dati:

```
df %>%
  ggplot(aes(x = mom_iq, y = kid_score)) +
  geom_point() +
  geom_abline(
    intercept = fixef(m, robust = TRUE)[1, 1],
    slope = fixef(m, robust = TRUE)[2, 1],
    size = 1/3
  ) +
  annotate(
    geom = "text",
    x = 130, y = 30,
    label = expression(y == 25.7 + 0.61 * x)
  ) +
  labs(
    subtitle = "Dati e retta di regressione",
    x = "x",
    y = "y"
  )
```

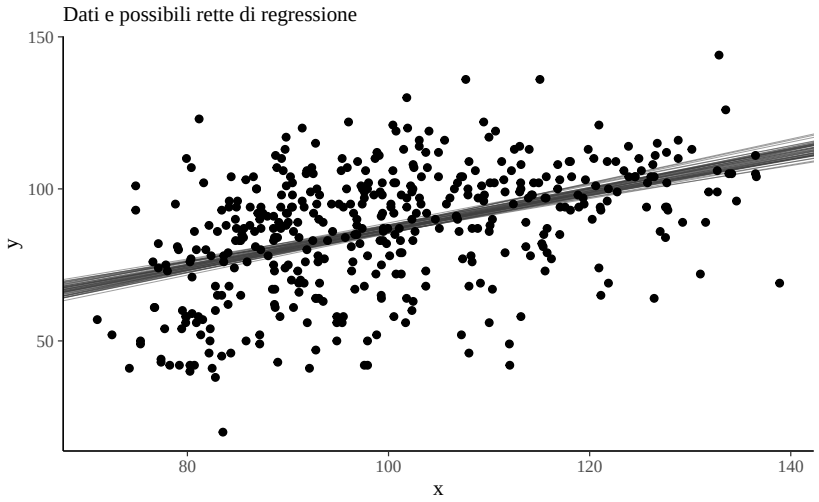


Usiamo ora la funzione `brms::posterior_samples()` per estrarre molti campioni dalla distribuzione a posteriori del modello `m`. In questo modo otteniamo un vettore di valori per ciascuno dei tre parametri, i quali, in questo output sono chiamati `b_Intercept`, `b_mom_iq` e `sigma`. Abbiamo quindi usato `slice_sample()` per ottenere un sottoinsieme casuale di 50 righe. Per semplicità, qui ne stampiamo solo 5.

```
set.seed(8)

posterior_samples(m) %>%
  slice_sample(n = 5)
#>   b_Intercept b_mom_iq sigma Intercept  lp__
#> 1        18.7    0.671  19.2      85.8 -1883
#> 2        20.4    0.663  17.9      86.7 -1881
#> 3        33.3    0.546  18.5      87.8 -1882
#> 4        25.5    0.620  18.3      87.5 -1881
#> 5        29.0    0.577  18.2      86.7 -1881
```

Possiamo interpretare i valori `b_Intercept`, `b_mom_iq` come un insieme di valori credibili per i parametri a e b. Dati questi valori credibili per i parametri del modello di regressione, possiamo aggiungere al diagramma a dispersione 50 stime possibili della retta di regressione, alla luce dei dati osservati.

```
set.seed(8)

posterior_samples(m) %>%
  slice_sample(n = 50) %>%

  ggplot() +
  geom_abline(
    aes(intercept = b_Intercept, slope = b_mom_iq),
        size = 1/4, alpha = 1/2, color = "grey25") +
  geom_point(
    data = df,
    aes(x = mom_iq, y = kid_score)
  ) +
  labs(
    subtitle = "Dati e possibili rette di regressione",
    x = "x",
    y = "y"
  )
```



I minimi quadrati o la massima verosimiglianza trovano i parametri che meglio si adattano ai dati (secondo un criterio prestabilito), ma senza altrimenti vincolare o guidare l'adattamento. Ma di solito abbiamo informazioni preliminari sui parametri del modello. L'inferenza bayesiana produce un compromesso tra informazioni a priori e i dati, moltiplicando la verosimiglianza con una distribuzione a priori che codifica probabilisticamente le informazioni esterne sui parametri. Il prodotto della verosimiglianza $p(y \mid a, b, \sigma)$ e della distribuzione a priori è chiamato *distribuzione a posteriori* e, dopo aver visto i dati, riassume la nostra credenza sui valori dei parametri.

La soluzione dei minimi quadrati fornisce una stima puntuale dei coefficienti che producono il miglior adattamento complessivo ai dati. Per un modello bayesiano, la corrispondente stima puntuale è la moda a posteriori, la quale fornisce il miglior adattamento complessivo ai dati e alla distribuzione a priori. La stima dei minimi quadrati o di massima verosimiglianza è la moda a posteriori corrispondente al modello bayesiano che utilizza una distribuzione a priori uniforme.

Ma non vogliamo solo una stima puntuale; vogliamo anche una misura dell'incertezza della stima. La figura precedente fornisce, in forma grafica, una descrizione di tale incertezza.

Gli intervalli di credibilità al 95% si ottengono nel modo seguente:

```
print(m, robust = TRUE)
#>  Family: gaussian
#>   Links: mu = identity; sigma = identity
#> Formula: kid_score ~ mom_iq
#>    Data: df (Number of observations: 434)
#>   Draws: 4 chains, each with iter = 1000; warmup = 0; thin = 1;
#>          total post-warmup draws = 4000
#>
#> Population-Level Effects:
#>           Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
#> Intercept    25.83      5.80    14.44    37.47 1.00     3329     3005
#> mom_iq        0.61      0.06     0.49     0.72 1.00     3310     3009
#>
#> Family Specific Parameters:
#>       Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
#> sigma    18.26      0.61    17.13    19.53 1.00     3824     3105
#>
#> Draws were sampled using sample(hmc). For each parameter, Bulk_ESS
```

```
#> and Tail_ESS are effective sample size measures, and Rhat is the potential
#> scale reduction factor on split chains (at convergence, Rhat = 1).
```

In alternativa, è possibile usare la funzione `posterior_interval()`:

```
posterior_interval(m)
#>                  2.5%     97.5%
#> b_Intercept    14.445    37.471
#> b_mom_iq        0.495     0.722
#> sigma          17.132    19.529
#> Intercept      85.119    88.563
#> lp__        -1885.190 -1880.580
```

Bibliografia

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

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

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.