Bike

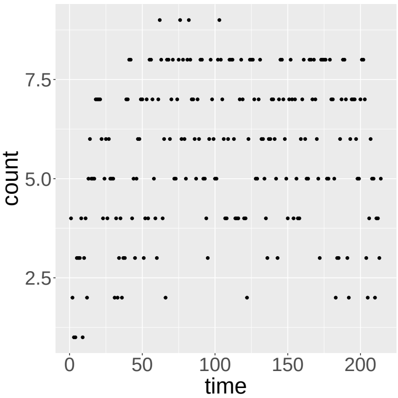
Vers. 1.1.1

Gianluca Mastrantonio

email: gianluca.mastrantonio@polito.it

Dataset Bike

Abbiamo un dataset in cui sono state contate, in migliaia, il numero di bici che sono passata su un ponte di New York per 7 mesi. Un grafico dei conteggi, rispetto al tempo, lo trovate qui sotto



I dati

Definiamo come Y_t^* la variabile aleatoria che rappresenta il conteggio al tempo t, e per semplicità definiamo

$$Y_t = \frac{Y_t^*}{1000}$$

Nel caso di conteggi, un modello standard è

$$Y_t^* \sim P(\exp(\mathbf{x}_t \boldsymbol{\beta}))$$

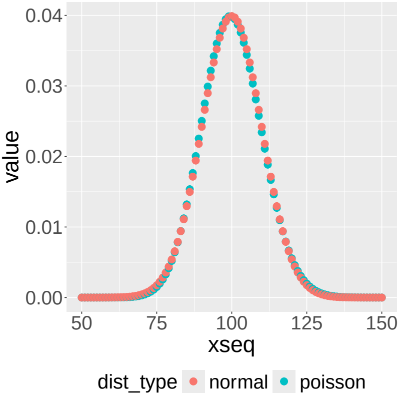
dove l'esponenziale si usa per garantire che il parametro della poisson sia positivo.
Come prima approssimazione, visto che i conteggi sono abbastanza elevati e non ci aspettiamo valori attesi vicini allo zero, possiamo utilizzare

$$Y_t^* \sim P(\mathbf{x}_t \boldsymbol{\beta})$$

(Attenzione: questo lo facciamo solo per semplificarci la vita su questo esempio, ma in generale sarebbe meglio non farlo.)

I dati

Come seconda approssimazione, visto che i conteggi sono molto elevati, possiamo approssimare la Poisson con una normale, visto che sono praticamente identiche se la normal ha la stessa media e varianza delle Poisson



I dati

Come modello finale assumiamo

$$Y_t^* \sim N(\mathbf{x}_t\boldsymbol{\beta}, \sigma^2)$$

dove la terza approssimazione sta nel fatto che la varianza non è uguale alla media, oppure, in maniera equivalente

$$Y_t \sim N(\mathbf{x}_t\boldsymbol{\beta}, \sigma^2)$$

Naturalmente i parametri in queste due normali, non hanno la stessa interpretazione.

Purtroppo noi non abbiamo osservato il vero conteggio, ma solo il conteggio in termine di migliaia. Facciamo l'assunzione che i numeri registrati siano l'arrotondamento al migliaio più vicino. Questo significa che se indichiamo con S_t il dato registrato, abbiamo che

$$S_t = \lfloor Y_t \rceil$$

e

$$Y_t = S_t + U_t$$

con $U_t \in (-0.5, 0.5]$.

Il modello

Il modello che cerchiamo è

$$f(\boldsymbol{\beta}, \sigma^2 | \mathbf{s}) \propto f(\mathbf{s} | \boldsymbol{\beta}, \sigma^2) f(\boldsymbol{\beta}, \sigma^2)$$

dove le prior, assumendo indipendenza, sono

$$\boldsymbol{\beta} \sim N(\mathbf{M}.\mathbf{V})$$

e

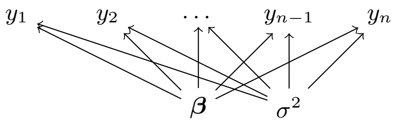
$$\sigma^2 \sim IG(a, b)$$

Dobbiamo quindi trovare la distribuzione dei dati (verosimiglianza).
Abbiamo che

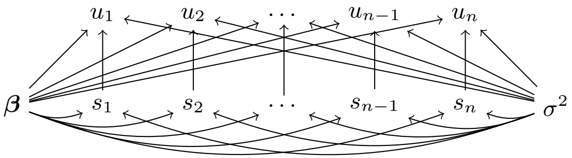
$$P(S_t = c) = \int_{c-0.5}^{c+0.5} f(y_t | \boldsymbol{\beta}, \sigma^2) dy = \Phi(c + 0.5 | \mathbf{x}_t \boldsymbol{\beta}, \sigma^2) - \Phi(c - 0.5 | \mathbf{x}_t \boldsymbol{\beta}, \sigma^2)$$

dove Φ sono le cumulate della normale.
Visto che le Y_t cono condizionatamente indipendenti, lo sono anche le S_t, questo lo potete vedere facendo i calcoli, ma anche dal grafo.

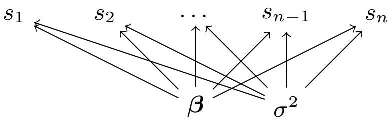
Il modello



che in termini di S_t e U_t si scrive come



e marginalizzando abbiamo



Il modello

Abbiamo quindi che il numeratore della a posteriori è

$$f(\mathbf{s}|\boldsymbol{\beta},\sigma^2)f(\boldsymbol{\beta},\sigma^2) = \left(\prod_{t=1}^{n} f(s_t|\boldsymbol{\beta},\sigma^2)\right) f(\boldsymbol{\beta},\sigma^2)$$

con

$$\begin{aligned}
f(s_t|\boldsymbol{\beta},\sigma^2) &= \Phi(s_t + 0.5|\mathbf{x}_t\boldsymbol{\beta},\sigma^2) - \Phi(s_t - 0.5|\mathbf{x}_t\boldsymbol{\beta},\sigma^2) \\
f(\boldsymbol{\beta}) &= (2\pi)^{-\frac{p}{2}}|\mathbf{V}|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\boldsymbol{\beta}-\mathbf{M})^T\mathbf{V}^{-1}(\boldsymbol{\beta}-\mathbf{M})\right) \\
f(\sigma^2) &= \frac{b^a}{\Gamma(a)}(\sigma^2)^{-(a+1)} \exp\left(-\frac{b}{\sigma^2}\right)
\end{aligned}$$

Le full conditional non esistono in forma chiusa, anche perchè la cumulata della normale non esiste in forma chiusa (si valuta algoritmicamente). Vedremo che in questo caso si potrebbe usare un passo Metropolis, ma, almeno in questo caso, abbiamo una diverse alternativa.

Il modello

Spesso, in statistica Bayesiana, si introducono altra variabili nel modello, che possono avere un'interpretazione o no. Queste aumentano lo spazio dei parametri, ma se lo facciamo in maniera furba, possiamo semplificare l'inferenza. Queste variabili vengono chiamata **variabili latenti** e gli algoritmi che si basano su queste variabili sono algoritmi di **data augmentation**.

Nel nostro caso possiamo introdurre le U_t e lavorare con la a-posteriori

$$f(\boldsymbol{\beta}, \sigma^2, \mathbf{u}|\mathbf{s}) \propto f(\mathbf{s}|\mathbf{u}, \boldsymbol{\beta}, \sigma^2) f(\mathbf{u}|\boldsymbol{\beta}, \sigma^2) f(\boldsymbol{\beta}, \sigma^2) = f(\mathbf{s}, \mathbf{u}|\boldsymbol{\beta}, \sigma^2) f(\boldsymbol{\beta}, \sigma^2)$$

Conosciamo la distribuzione marginale di S_t, e dobbiamo ricavare solamente la condizionata $U_t|S_t$ (Attenzione per semplificare un po' la notazione, tolgo il condizionamento a σ^2 e $\boldsymbol{\beta}$). Abbiamo che, se $u_t \in (-0.5, 0.5]$, allora

$$P(U_t \leq u_t|S_t = s_t) = P(s_t - 0.5 < Y_t \leq s_t + u_t|S_t = s_t) =$$

$$\frac{P(s_t - 0.5 < Y_t \leq s_t + u_t, S_t = s_t)}{P(S_t = s_t)} = \frac{P(s_t - 0.5 < Y_t \leq s_t + u_t)}{P(S_t = s_t)}$$

$$\frac{\Phi(s_t + u_t|\mathbf{x}_t\boldsymbol{\beta}, \sigma^2) - \Phi(s_t - 0.5|\mathbf{x}_t\boldsymbol{\beta}, \sigma^2)}{\Phi(s_t + 0.5|\mathbf{x}_t\boldsymbol{\beta}, \sigma^2) - \Phi(s_t - 0.5|\mathbf{x}_t\boldsymbol{\beta}, \sigma^2)}$$

Il modello

Per calcolare la densità possiamo derivare e abbiamo che

$$f(u_t|s_t) \propto \phi(s_t + u_t|\mathbf{x}_t\boldsymbol{\beta}, \sigma^2)$$

e visto che la densità deve integrare a 1, e che $u_t \in (-0.5, 0.5]$, allora

$$f(u_t|s_t, \boldsymbol{\beta}, \sigma^2) = \frac{\phi(s_t + u_t|\mathbf{x}_t\boldsymbol{\beta}, \sigma^2)}{\Phi(s_t + 0.5|\mathbf{x}_t\boldsymbol{\beta}, \sigma^2) - \Phi(s_t - 0.5|\mathbf{x}_t\boldsymbol{\beta}, \sigma^2)} =$$

$$\frac{(2\pi\sigma^2)^{-0.5} \exp\left(-\frac{(s_t + u_t - \mathbf{x}_t\boldsymbol{\beta})^2}{2\sigma^2}\right)}{\Phi(s_t + 0.5|\mathbf{x}_t\boldsymbol{\beta}, \sigma^2) - \Phi(s_t - 0.5|\mathbf{x}_t\boldsymbol{\beta}, \sigma^2)} =$$

$$\frac{(2\pi\sigma^2)^{-0.5} \exp\left(-\frac{(u_t + s_t - \mathbf{x}_t\boldsymbol{\beta})^2}{2\sigma^2}\right)}{\Phi(s_t + 0.5|\mathbf{x}_t\boldsymbol{\beta}, \sigma^2) - \Phi(s_t - 0.5|\mathbf{x}_t\boldsymbol{\beta}, \sigma^2)} =$$

e quindi

$$U_t|s_t, \boldsymbol{\beta}, \sigma^2 \sim TN(\mathbf{x}_t\boldsymbol{\beta} - s_t, \sigma^2)$$

Il modello

Questo ci dice che

$$f(s_t, u_t|\boldsymbol{\beta}, \sigma^2) =$$

$$\frac{\phi(s_t + u|\mathbf{x}_t\boldsymbol{\beta}, \sigma^2)}{\Phi(s_t + 0.5|\mathbf{x}_t\boldsymbol{\beta}, \sigma^2) - \Phi(s_t - 0.5|\mathbf{x}_t\boldsymbol{\beta}, \sigma^2)} \left(\Phi(s_t + 0.5|\mathbf{x}_t\boldsymbol{\beta}, \sigma^2) - \Phi(s_t - 0.5|\mathbf{x}_t\boldsymbol{\beta}, \sigma^2)\right)$$

$$\phi(s_t + u_t|\mathbf{x}_t\boldsymbol{\beta}, \sigma^2) = \phi(y_t|\mathbf{x}_t\boldsymbol{\beta}, \sigma^2)$$

dove ora

$$y_t = s_t + u_t$$

va visto come funzione dell'osservata s_t e della latente u_t

Il modello

Adesso abbiamo che la a posteriori è proporzionale a

$$f(\mathbf{s}, \mathbf{u}|\boldsymbol{\beta}, \sigma^2) f(\boldsymbol{\beta}, \sigma^2)$$

con

$$f(\mathbf{s}, \mathbf{u}|\boldsymbol{\beta}, \sigma^2) = (2\pi\sigma^2)^{-\frac{n}{2}} \exp\left(-\frac{1}{2\sigma^2}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^T(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})\right)$$

$$f(\boldsymbol{\beta}) = (2\pi)^{-\frac{p}{2}}|\mathbf{V}|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\boldsymbol{\beta} - \mathbf{M})^T\mathbf{V}^{-1}(\boldsymbol{\beta} - \mathbf{M})\right)$$

$$f(\sigma^2) = \frac{b^a}{\Gamma(a)}(\sigma^2)^{-(a+1)} \exp\left(-\frac{b}{\sigma^2}\right)$$

Le full conditional di β e σ^2 sono le stesse del caso regressivo standard

$$\boldsymbol{\beta}|\mathbf{y}, \sigma^2 \sim N_p(\mathbf{M}_p, \mathbf{V}_p)$$

$$\sigma^2|\boldsymbol{\beta}, \mathbf{y} \sim IG\left(\frac{n}{2} + a, \frac{1}{2}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^T(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) + b\right)$$

con

$$\mathbf{V}_p = (\frac{1}{\sigma^2}\mathbf{X}^T\mathbf{X} + \mathbf{V}^{-1})^{-1}$$
$$\mathbf{M}_p = \mathbf{V}_p(\frac{1}{\sigma^2}\mathbf{X}^T\mathbf{y} + \mathbf{V}^{-1}\mathbf{M})$$

Adesso dobbiamo anche campionare ogni U_t, ma la full conditional, che è la distribuitone di

$$U_t|s_t, \boldsymbol{\beta}, \sigma^2$$

l'abbiamo già calcolata prima e è

$$U_t|s_t, \boldsymbol{\beta}, \sigma^2 \sim TN(\mathbf{x}_t\boldsymbol{\beta} - s_t, \sigma^2)$$

Il modello

Implementare un modello basandoci su

$$f(\boldsymbol{\beta}, \sigma^2|\mathbf{s}) = \frac{f(\mathbf{s}|\boldsymbol{\beta}, \sigma^2) f(\boldsymbol{\beta}, \sigma^2)}{f(\mathbf{s})}$$

o

$$f(\boldsymbol{\beta}, \sigma^2, \mathbf{u}|\mathbf{s}) = \frac{f(\mathbf{s}, \mathbf{u}|\boldsymbol{\beta}, \sigma^2) f(\boldsymbol{\beta}, \sigma^2)}{f(\mathbf{s})}$$

produce gli stessi risultati perchè il primo si può vedere come marginalizzazione del secondo rispetto alle u_t. Oltretutto, in entrambi i casi, quando guardiamo i campioni dei parametri, stiamo vedendo i campioni dalle marginali

$$f(\boldsymbol{\beta}|\mathbf{s}) \qquad f(\sigma^2|\mathbf{s})$$

Nota1 anche se campiono dal modello marginale, posso riottenere, dopo il model fitting, i valori di u_t, visto che

$$f(\boldsymbol{\beta}, \sigma^2, \mathbf{u}|\mathbf{s}) = f(\mathbf{u}|\boldsymbol{\beta}, \sigma^2, \mathbf{s}) f(\boldsymbol{\beta}, \sigma^2|\mathbf{s})$$

e quindi campioni da $f(\boldsymbol{\beta}, \sigma^2, \mathbf{u}|\mathbf{s})$ si possono ottenere

Il modello

- campionando da $f(\boldsymbol{\beta}, \sigma^2|\mathbf{s})$ (ma ho i campioni dalla a posteriori che trovo con MCMC)
- campiono da $f(\mathbf{u}|\boldsymbol{\beta}, \sigma^2, \mathbf{s}) = \prod_{i=1}^{n} f(u_t|\boldsymbol{\beta}, \sigma^2, s_t)$,

Nota2 Se per ogni campioni u_t^b calcolo

$$y_t^b = s_t + u_t^b$$

questi sono campioni dalla a-posteriori

$$f(y_t|s_t)$$

perchè posso vederli come una trasformazione di variabili

$$f_y(y_t|s_t) = f_u(u_t|s_t)|\frac{du_t}{dy_t}|$$

in generale questo è vero sempre:

Il modello

Se avete una a posteriori

$$f(\boldsymbol{\theta}|\mathbf{s})$$

e per ogni campione a posteriori calcolate la stessa quantità (in termine di forma funzionale)

$$w^b = g(\mathbf{s}, \boldsymbol{\theta}^b)$$

quello che ottenete sono dei campioni dalla a posteriori di w $(f(w|\mathbf{s}))$.
Quindi, anche se non abbiamo veramente osservato y_t, possiamo averne

- una stima della sua distribuzione
- una stima puntuale (media a posteriori)
- definire un intervallo di valori probabili

Risultati

Implementiamo il modello (lo trovate in cartella) in cui, come covariate usiamo il giorno della settimana e time. Vediamo prima alcune statistiche che ci propone il pacchetto R "coda"

	Mean	SD	Naive SE	Time-series SE
(Intercept)	4.99	0.34	0.01	0.01
weekdayMonday	0.27	0.40	0.01	0.01
weekdaySaturday	-1.21	0.38	0.01	0.01
weekdaySunday	-1.80	0.39	0.01	0.01
weekdayThursday	1.03	0.39	0.01	0.01
weekdayTuesday	1.00	0.41	0.01	0.01
weekdayWednesday	1.47	0.40	0.01	0.01
time	0.01	0.00	0.00	0.00
sigma2	2.32	0.24	0.01	0.01

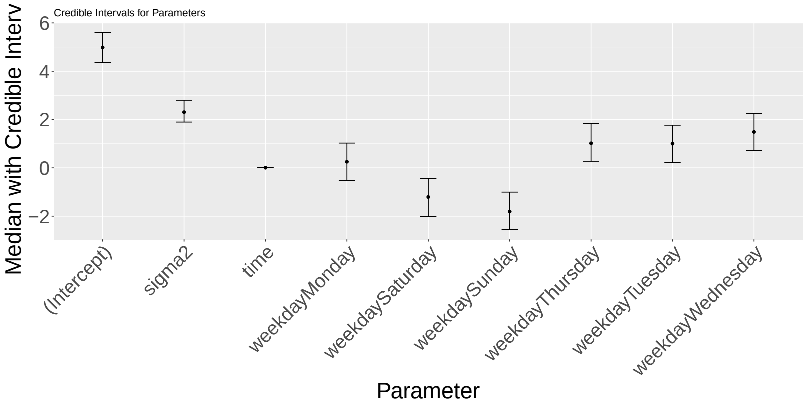
Risultati

E i quantili

	2.5%	25%	50%	75%	97.5%
(Intercept)	4.33	4.78	4.97	5.22	5.70
weekdayMonday	-0.52	0.02	0.27	0.54	1.01
weekdaySaturday	-1.97	-1.46	-1.20	-0.96	-0.48
weekdaySunday	-2.57	-2.05	-1.79	-1.53	-1.06
weekdayThursday	0.23	0.78	1.03	1.29	1.80
weekdayTuesday	0.23	0.73	1.01	1.28	1.80
weekdayWednesday	0.68	1.21	1.48	1.74	2.23
time	0.00	0.00	0.01	0.01	0.01
sigma2	1.90	2.15	2.30	2.48	2.83

Risultati

e i quantile in versione grafica



Ricordate che in questo caso l'intercetta $beta_{\text{intercetta}}$ rappresenta l'effetto della classe mancante (Friday), e i valori associati ai giorni della settimana sono la differenza rispetto al corner, e la indico come, per esempio, $\beta_{\text{diff monday}}$. Possiamo

Risultati

- ottenere campioni dell'effetto "puro" del lunedì come

$$\beta^b_{\text{monday}} = \beta^b_{\text{intercetta}} + \beta^b_{\text{diff monday}}$$

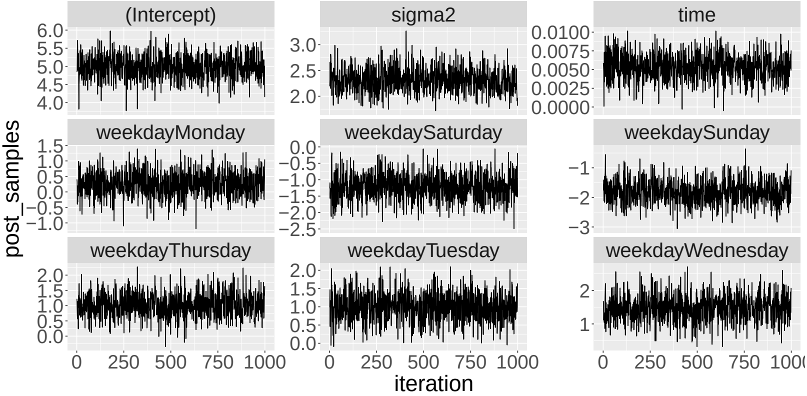
- ottenere campioni della differenza tra lunedì e martedì come

$$\beta^b_{\text{intercetta}} + \beta^b_{\text{diff monday}} - (\beta^b_{\text{intercetta}} + \beta^b_{\text{difftuesday}}) = \beta^b_{\text{diff monday}} - \beta^b_{\text{difftuesday}}$$

- qualsiasi altra differenza/trasformazione, usando direttamente i campioni

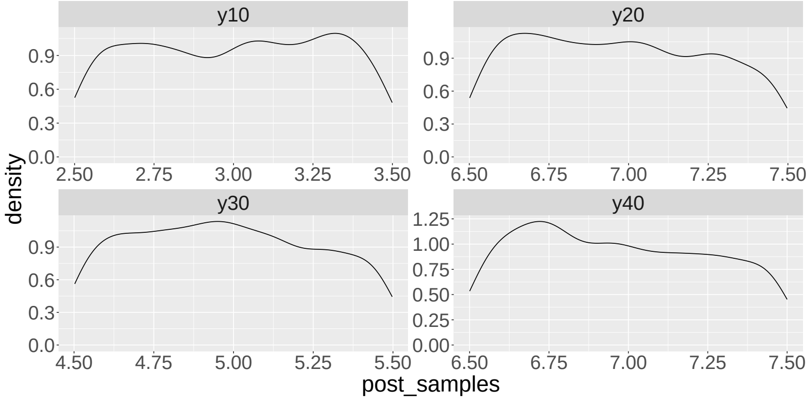
Risultati

Possiamo anche vedere le catene



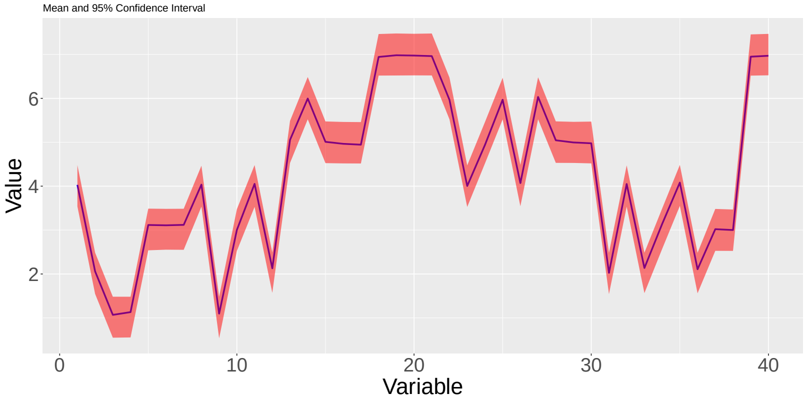
Risultati

Mostriamo alcune distribuzioni di y a posteriori



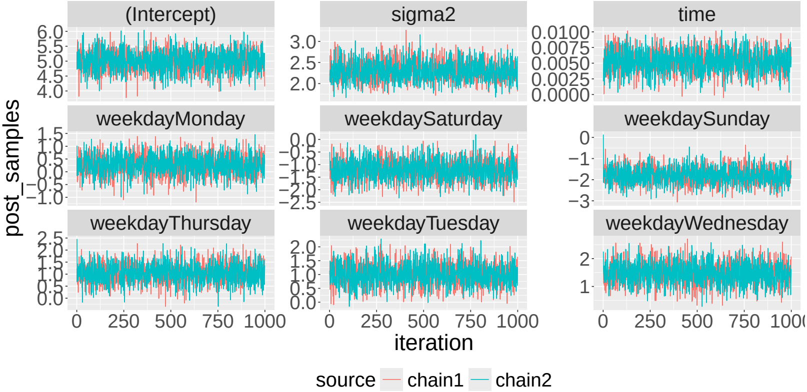
Risultati

Oppure la media a posteriori (linea) rispetto al tempo e l'intervallo di credibilità al 95% (shaded area) di una porzione dei dati



Catene multiple

Valutare se l'algoritmo è arrivato a convergenza non è mai facile. Un modo piuttosto semplice, anche se dispendioso a livello computazionale, è di far partire diverse MCMC da valori iniziali molto diversi, e verificare se arrivano allo stesso punto. Se questo si verifica, siamo abbastanza certi che siamo arrivati alla distribuzione stazionaria. I due colori sotto rappresentano le due catene, che sono arrivate nello stesso spazio



Model prediction and Model performance

Per valutare la bontà del modello abbiamo diverse scelte.

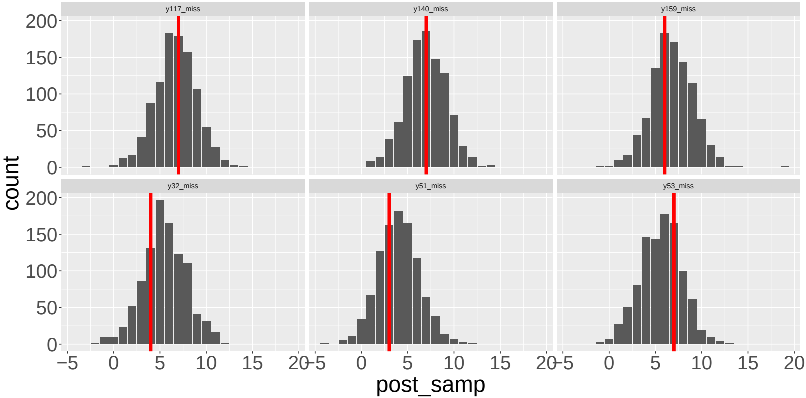
- confrontare una stima a posteriori di s_t, per esempio $\mathbf{x}_t\boldsymbol{\beta}$, con s_t, ma non sempre ha senso e poi confronti in-sample non danno risultati sempre soddisfacente
- confrontare le previsioni di s_t, ottenute con un modello che non contiene s_t, e confrontarle con il valore vero (**cross validation**). In dettaglio, potremmo eliminare delle osservazioni $s_{i,miss}$ (il set totale dei missing è $\mathbf{s}_{miss}$, composto da n_{miss} data), e stimare il modello con $\mathbf{s}_{obs}$. Poi, tramite MC, otteniamo campioni dalla loro a posteriori

$$f(s_{i,miss}|\mathbf{s}_{obs}) = \int\int\int f(s_{i,miss}, u_{i,miss}|\boldsymbol{\beta}, \sigma^2, \mathbf{s}_{obs}) f(\boldsymbol{\beta}, \sigma^2|\mathbf{s}_{obs}) du_{i,miss} d\boldsymbol{\beta} d\sigma^2$$

LA distribuzione $f(s_{i,miss}|\mathbf{s}_{obs})$ si chiama **distribuzione predittiva**

Alcuni risultati, usando il secondo punto, sono mostrati nella figura di sotto

Model prediction and Model performance



Model prediction and Model performance

In caso avessimo più specificazioni del modello da testare per decidere quale delle due è migliore, usando il modello con i miss, potremmo calcolare

- CRPS per osservazione:

$$CRPS_i = \frac{\sum_{b=1}^{B} |s_{i,miss} - s_{i,miss}^b|}{B} - \frac{1}{2B} \frac{\sum_{b=1}^{B} \sum_{h=1}^{B} |s_{i,miss}^b - s_{i,miss}^h|}{B}$$

- MSE totale:

$$MSE = \frac{\sum_{i=1}^{n_{miss}} (s_{i,miss} - \hat{s}_{i,miss})^2}{B}$$

  dove $\hat{s}_{i,miss}$ è una stima puntuale dalla predittiva, per esempio la media a posteriori

Se invece vogliamo usare il modello completo con tutti i dati, possiamo calcolare, per esempio

Model prediction and Model performance

- AIC, che in questo caso è

$$AIC = -2 log f(\mathbf{s}|\hat{\boldsymbol{\beta}}, \hat{\sigma}^2) + 2(p+1) =$$

$$-2\left(\Phi(s_t + 0.5|\mathbf{x}_t\hat{\boldsymbol{\beta}}, \hat{\sigma}^2) - \Phi(s_t - 0.5|\mathbf{x}_t\hat{\boldsymbol{\beta}}, \hat{\sigma}^2)\right) + 2(p+1)$$

  fate attenzione che i parametri non contengono le u_t, perchè abbiamo scritto la verosimiglianza usando le s_t. Se condizionassimo a le u, dovremmo mettere anche il numero delle u nel numero dei parametri. Però, quando mettiamo variabili latenti nei parametri, l'AIC non performa bene

- WAIC:

$$\text{WAIC} = -2\sum_{i=1}^{n}\log\left(\frac{1}{B}\sum_{b=1}^{B} f(s_i|\hat{\boldsymbol{\beta}}^b, \hat{\sigma}^{2,b})\right) + 2\sum_{i=1}^{n}\text{Var}_{\text{posterior}}\left(\log f(s_i|\boldsymbol{\beta}, \sigma^2)\right) =$$

$$-2\sum_{i=1}^{n}\log\left(\frac{1}{B}\sum_{b=1}^{B}\left(\Phi(s_t + 0.5|\mathbf{x}_t\hat{\boldsymbol{\beta}}^b, \hat{\sigma}^{2,b}) - \Phi(s_t - 0.5|\mathbf{x}_t\hat{\boldsymbol{\beta}}^b, \hat{\sigma}^{2,b})\right)\right) +$$

$$+2\sum_{i=1}^{n}\frac{\sum_{b=1}^{B}(\log f(s_i|\hat{\boldsymbol{\beta}}^b, \hat{\sigma}^{2,b}) - \sum_{h=1}^{B}\log f(s_i|\hat{\boldsymbol{\beta}}^b, \hat{\sigma}^{2,b})/B)^2}{B}$$

Metropolis

Se per qualche motivo, non possiamo simulare U_t (o un'altra variabile) dalla full conditional direttamente, allora possiamo usare un passo Metropolis. In questo caso potremmo

- Usare una proposta
$$U(max(u_t^b - h, -0.5), min(u_t^b + h, -0.5))$$
che ha densità
$$\frac{1}{min(u_t^b + h, -0.5) - max(u_t^b - h, -0.5)}$$
-
$$U(u_t^b - h, u_t^b + h)$$
che ha densità uguale per la proposta e l'accettato, e quindi scompare nel rapporto metropolis. In questo caso, se andiamo fuori da $(-0.5, 0.5]$ dobbiamo rifiutare

Metropolis

- $$N(u_t^b, \tau^2)$$

  che ha densità uguale per la proposta e l'accettato, e quindi scompare nel rapporto metropolis. In questo caso, se andiamo fuori da $(-0.5, 0.5]$ dobbiamo rifiutare
- Fare un cambio di variabili e lavorare con

  $$r_t = \log\left(\frac{u_t + 0.5}{0.5 - U_t}\right) \in \mathbb{R}$$

  e quindi con la full conditional

  $$f_r(r_t|\mathbf{s}, \boldsymbol{\beta}, \sigma^2) = f_u(u_t|\mathbf{s}, \boldsymbol{\beta}, \sigma^2)|\frac{du_t}{dr_t}|$$

  e usare una proposta

  $$N(u_t^b, \tau^2)$$

  che ha densità uguale per la proposta e l'accettato, e quindi scompare nel rapporto metropolis.

Nel caso delle proposte normali, dobbiamo in qualche modo decidere/trovare τ^2, e in quella uniforme h.

Simulazioni

Quando scrivete del codice è sempre il caso di verificare che funzioni. Un modo, relativamente facile, è

- simulare dei dati, applicare l'algoritmo e vedere se il 95% dei parametri (approssimativamente) sia dentro l'intervallo a posteriori del 95%. (test generale)
- settare tutti i parametri al valore vero, e campionarne solo uno, o un set (valutare se il sampling di un parametro funziona). Poi dovete vedere se il 95% dei parametri (approssimativamente) sia dentro l'intervallo a posteriori del 95%.

Questo ha senso se le prior sono weakly informative Nel nostro caso, per esempio ho simulato dei dati col seguente codice

Simulazioni

```
p <- 3
n <- 1000
beta <- runif(p, 0, 10)
X <- matrix(1, nrow = n, ncol = p)
sigma2 <- 0.2
for (icol in 2:p)
{
  X[, icol] <- runif(n, -5, 5)
}
y <- rnorm(n, X %*% matrix(beta, ncol = 1), sigma2^0.5)
s <- round(y)
u <- y - s
```

Simulazioni

e poi possiamo vedere le catene con la linea orizzontale che rappresenta il valore vero