Università degli Studi di Padova

Dipartimento di Scienze Statistiche
Corso di Laurea Magistrale in
Scienze Statistiche

Modelli bayesiani nonparametrici: applicazioni al settore assicurativo

Relatore Prof. Antonio Canale
Dipartimento di Scienze Statistiche

Laureanda: Laura D'Angelo
Matricola N. 1131343

Anno Accademico 2016/2017

Indice

Introduzione

Nell'ambito delle assicurazioni sulla vita, una particolare categoria è costituita dalle assicurazioni collettive. Queste sono particolari polizze stipulate da un unico contraente per assicurare un gruppo di individui: il caso più tipico è quello di un datore di lavoro che assicura i dipendenti. Contrariamente alle assicurazioni sulla vita individuali, in questo caso il contraente può verificare se il contratto che ha stipulato gli è conveniente: sul lungo termine, infatti, può verificare se i risarcimenti ottenuti sono comparabili al premio pagato e, se così non fosse, potrebbe pensare di cambiare polizza. Per questo motivo, l'assicurazione è interessata a stimare nel modo più preciso possibile il numero di richieste di risarcimento di un particolare gruppo. Una difficoltà in questo ambito sorge tuttavia dall'impossibilità, in genere per questioni economiche, di registrare tutte le caratteristiche che influiscono sul rischio complessivo di un particolare gruppo, comportando la presenza di una grande eterogeneità non osservata. Un modello adeguato deve, quindi, tenere in considerazione questa eterogeneità tra i gruppi ma, allo stesso tempo, deve permettere anche di identificare i gruppi simili per definire un numero limitato di categorie di rischio.

In questa tesi, per rispondere a queste problematiche, si propone un approccio bayesiano nonparametrico basato su un modello mistura che tenga conto della particolare natura dei dati, costituiti da conteggi. In generale, una mistura nonparametrica è una mistura di *kernel* parametrici, in cui la distribuzione misturante è anch'essa casuale e specificata per mezzo di particolari processi stocastici. L'obiettivo di questa tesi è di confrontare i risultati che si ottengono a partire da diverse scelte sia per il *kernel* sia per il processo stocastico. In particolare, per il processo stocastico si considereranno i processi di Dirichlet e di Pitman-Yor; per la scelta del *kernel*, poiché dipende dalla natura dei dati, si confronteranno i *kernel* Poisson e *Rounded Gaussian*.

Nel Capitolo 1 si introdurranno le nozioni teoriche e computazionali necessarie alla specificazione del modello; nel Capitolo 2 si condurrà uno studio di simulazione per confrontare i risultati delle diverse formulazioni; nel Capitolo 3, infine, si applicherà la metodologia proposta ai dati in esame.

Capitolo 1

Metodi bayesiani nonparametrici

1.1 Modelli bayesiani nonparametrici

L'inferenza bayesiana si basa sul teorema di Bayes, che lega la probabilità iniziale (a priori) e la verosimiglianza per ottenere la distribuzione a posteriori: questo è il meccanismo su cui si basa l'acquisizione di nuova conoscenza, ovvero l'aggiornamento delle opinioni iniziali sulla base degli eventi osservati (Cifarelli e Muliere, 1989).

Per formalizzare questo meccanismo occorre la definizione del modello statistico, che riassume la conoscenza e l'incertezza riguardo al fenomeno. Effettuato un esperimento $\mathcal{E}$, questo può infatti produrre diversi risultati, di cui solo uno è effettivamente osservato: $\mathcal{E}$ è, infatti, un singolo punto dello spazio Ω di tutti i possibili esiti. Se su Ω si considera una σ-algebra $\mathcal{F}$, sullo spazio $(\Omega, \mathcal{F})$ si può definire una famiglia di misure di probabilità $\mathcal{P}$, che costituisce l'obiettivo dell'inferenza. Un modello statistico è costituito dalla collezione di misure probabilità $P \in \mathcal{P}$, che vengono in genere indicizzate da un parametro $\theta \in \Theta$, ovvero:

$$\mathcal{P} = \{P_\theta,\ \theta \in \Theta\}.$$

Se sullo spazio $(\Omega, \mathcal{F}, \mathcal{P})$ si considera una variabile aleatoria X a valori in $\mathcal{X}$, il modello statistico può essere equivalentemente espresso come $\{p(\cdot|\theta), \theta \in \Theta\}$, dove p è l'ignota funzione di probabilità su $\mathcal{X}$.

Lo spazio Θ è detto spazio parametrico, e permette di identificare univocamente gli elementi di $\mathcal{P}$. Se Θ è un sottoinsieme dello spazio euclideo $\mathbb{R}^d$, con $d < \infty$, il modello è detto parametrico; altrimenti è detto nonparametrico.

La conoscenza a disposizione prima di osservare $\mathcal{E}$ viene espressa per mezzo di una densità su Θ, $\pi(\theta)$, che, grazie al teorema di Bayes, viene

modificata sulla base dell'osservazione di $(X_1, \ldots, X_n)$ come:

$$\pi(\theta|X_1, \ldots, X_n) = \frac{\prod_{i=1}^n p(x_i|\theta)\pi(\theta)}{\int_\Theta \prod_{i=1}^n p(x_i|\theta)\pi(\theta)d\theta}.$$

La distribuzione a priori e la funzione di probabilità $p(\cdot|\theta)$ non sono elementi sconnessi: insieme riassumono la conoscenza sulla distribuzione che ha generato i dati, in assenza di osservazioni. La probabilità a priori esprime, quindi, una distribuzione sullo spazio delle distribuzioni di probabilità: se si assume per p, per esempio, una normale, questa implica, insieme a π, una distribuzione sullo spazio di tutte le normali di media e varianza non note.

Si entra nel contesto dei modelli nonparametrici quando la forma della distribuzione di probabilità non è vincolata ad una particolare famiglia parametrica, ma si assume solo sia definita su un certo spazio $\mathcal{X}$. Il parametro ora corrisponde, quindi, a una distribuzione di probabilità su $\mathcal{X}$ e lo spazio parametrico diventa lo spazio delle distribuzioni $\mathcal{P}(\mathcal{X})$.

Un problema è quello di probabilizzare gli elementi di questo spazio, cioè definire delle distribuzioni su $\mathcal{P}(\mathcal{X})$, che ha dimensione infinita. Per fare ciò si fa uso di processi stocastici, che possono essere interpretati come distribuzioni su spazi di funzioni (Hjort et al., 2010).

1.2 Processi stocastici

Un processo stocastico è una collezione di variabili aleatorie definite su un comune spazio di probabilità $(\Omega, \mathcal{F}, \mathbb{P})$ indicizzate da una variabile $\alpha \in I$. L'insieme I degli indici del processo può essere un qualunque insieme numerabile o non numerabile, purché di cardinalità infinita.

In particolare, si supponga che per ogni $\alpha \in I$ esista una variabile aleatoria $X_\alpha : \Omega \to \mathbb{R}$ definita su $(\Omega, \mathcal{F}, \mathbb{P})$; la funzione $X : I \times \Omega \to \mathbb{R}$ definita da $X(\alpha, \omega) = X_\alpha(\omega)$ è detta processo stocastico (Clarke e Disney, 1985).

Un processo stocastico è quindi una funzione di due variabili: $\alpha \in I$ e $\omega \in \Omega$. Per α fissato, $X_\alpha(\omega)$ è una variabile aleatoria; per ω fissato, la funzione $\alpha \to X_\alpha(\omega)$ è la traiettoria associata a ω. Poiché ω è casuale, un processo stocastico può essere visto come una distribuzione su uno spazio di funzioni, le cui realizzazioni sono funzioni casuali (le traiettorie).

1.3 Il processo di Dirichlet

Il processo di Dirichlet (DP) è un processo stocastico le cui traiettorie sono distribuzioni di probabilità con probabilità uno. Fu introdotto da Ferguson

(1973, 1974) come soluzione al problema di definire una distribuzione a priori in contesto bayesiano nonparametrico, in cui lo spazio parametrico è di dimensione infinita ed è costituito da tutte le distribuzioni di probabilità definite su un certo spazio $\mathcal{X}$. In particolare, l'obiettivo era quello di definire una distribuzione a priori che ammettesse ampio supporto e allo stesso tempo fornisse una distribuzione a posteriori trattabile analiticamente.

Il processo di Dirichlet nasce come estensione al caso infinito-dimensionale della distribuzione Dirichlet e di questa mantiene, in particolare, due proprietà importanti.
La prima è la proprietà di coniugazione: come la distribuzione Dirichlet è coniugata al modello multinomiale, che specifica una distribuzione di probabilità arbitraria su un insieme finito di valori, allo stesso modo, il processo di Dirichlet è coniugato ad una arbitraria distribuzione di probabilità su un insieme di cardinalità infinita.
La seconda proprietà riguarda la distribuzione che deriva da raggruppamenti dei dati: se X è una variabile multinomiale di parametri $(\pi_1, \ldots, \pi_k)$ a valori in $\mathcal{X} = \{1, \ldots, k\}$ e $(\pi_1, \ldots, \pi_k) \sim Dir(\alpha_1, \ldots, \alpha_k)$, allora, se si considera una partizione $A_1, \ldots, A_m$ di $\mathcal{X}$, vale la distribuzione $(\sum_{j \in A_1} \pi_j, \ldots, \sum_{j \in A_m} \pi_j) \sim Dir(\sum_{j \in A_1} \alpha_j, \ldots, \sum_{j \in A_m} \alpha_j)$. Anche per il processo di Dirichlet vale una proprietà analoga rispetto all'aggregazione in classi dei dati, che ora appartengono all'insieme dei reali.

Il processo di Dirichlet è, quindi, un processo stocastico le cui realizzazioni sono misure di probabilità casuali, che genera distribuzioni di Dirichlet quando le osservazioni vengono raggruppate in un numero finito classi (Dey e Rao, 2005; Ghosh e Ramamoorthi, 2003; Hjort et al., 2010).

1.3.1 Proprietà

Il DP è parametrizzato da due quantità: α e P_0. Essi possono essere interpretati in funzione dei momenti del processo: P_0, nota anche come distribuzione di base, è il valore atteso, nel senso che, per ogni insieme misurabile $A \subset \mathcal{X}$, $\mathbb{E}[P(A)] = P_0(A)$; mentre il parametro di concentrazione α è funzione del reciproco della varianza, $Var(P(A)) = P_0(A)(1 - P_0(A))/(1 + \alpha)$. Scegliendo un DP come a priori per P, P_0 indica il "miglior" modello parametrico disponibile sulla base della conoscenza del processo, e α il peso dato a questa congettura: maggiore è α, minore è la varianza, e il processo avrà maggiore massa intorno alla media, mentre per $\alpha \to 0$ l'a priori tenderà ad essere non informativa.

Una proprietà importante di questo processo è che le sue realizzazioni sono distribuzioni discrete quasi certamente, anche quando la distribuzione di

base P_0 non lo è. Questo può apparire come un limite, poiché l'insieme delle distribuzioni con probabilità non nulla è costituito dalle sole distribuzioni discrete, tuttavia, il processo ha ampio supporto in un altro senso.

In generale, se P è una misura di probabilità, il supporto topologico di P è definito come il più piccolo insieme chiuso di misura 1 (rispetto a P). Si può mostrare che il supporto di un processo di Dirichlet $P \sim DP(\alpha, P_0)$, rispetto alla topologia debole, è formato da tutte le distribuzioni di probabilità P^* il cui supporto è contenuto nel supporto di P_0, ovvero:

$$supp(P) = \{P^* : supp(P^*) \subset supp(P_0)\}.$$

Se il supporto della distribuzione di base è tutta la retta reale, come per esempio nel caso della distribuzione normale, allora il supporto di P è costituito dall'insieme di tutte le distribuzioni di probabilità definite su $\mathbb{R}$: una sequenza di realizzazioni di un DP converge in distribuzione a qualunque distribuzione di probabilità definita su $\mathbb{R}$.

1.3.2 Distribuzione a posteriori

Sia $P \sim DP(\alpha, P_0)$. Poiché P è una distribuzione, è possibile considerare una sequenza i.i.d. $X_1, \ldots, X_n | P \sim P$. La distribuzione a posteriori che ne deriva è ancora un processo di Dirichlet:

$$P|X_1, \ldots, X_n \sim DP(\alpha + n, \frac{\alpha P_0 + \sum_{i=1}^{n} \delta_{X_i}}{\alpha + n})$$

dove δ_X indica la distribuzione di una variabile degenere in X.
Da questa equazione si nota come il processo di Dirichlet mantenga la proprietà di coniugazione del caso finito-dimensionale: ora le X_i non sono più realizzazioni da una distribuzione multinomiale $(p_1, \ldots, p_k)$ su k classi, ma sono generate da una generica distribuzione P sui reali.

Il valore atteso a posteriori è pari a:

$$\mathbb{E}[P|X_1, \ldots, X_n] = \frac{\alpha}{\alpha + n} \cdot P_0 + \frac{n}{\alpha + n} \cdot \big(\frac{1}{n} \sum_{i=1}^{n} \delta_{X_i}\big)$$

ovvero è una media pesata tra la media a priori e la distribuzione empirica di X. Per α fissato e $n \to +\infty$, il comportamento asintotico della media a posteriori è determinato da quello della distribuzione empirica.

1.3.3 Distribuzione predittiva: schema delle urne di Pólya

La distribuzione predittiva per una nuova realizzazione può essere espressa considerando un procedimento sequenziale detto delle urne di Pólya generalizzato.

A ogni passo si considera la distribuzione di una nuova estrazione condizionata alle precedenti, marginalizzata rispetto a P. La prima realizzazione sarà $X_1 \sim P_0$; la seconda osservazione X_2 sarà pari a X_1 con probabilità $1/(\alpha+1)$, oppure sarà un nuovo valore estratto da P_0 con probabilità $\alpha/(\alpha+1)$. Questo procedimento viene ripetuto per ogni osservazione successiva: il generico X_n sarà pari a un precedente valore X_i con probabilità $1/(\alpha+n-1)$ o a uno nuovo estratto da P_0 con probabilità $\alpha/(\alpha+n-1)$.

Poiché le realizzazioni di un DP sono distribuzioni discrete, in una serie di realizzazioni potranno comparire valori ripetuti. Se $(X_1^*, \ldots, X_K^*)$ sono i K valori distinti in $X_1, \ldots, X_{n-1}$, con frequenze rispettivamente $(n_1, \ldots, n_K)$, la distribuzione per X_n è pari a:

$$X_n | X_1, \ldots, X_{n-1} \sim \sum_{k=1}^{K} \frac{n_k}{\alpha+n-1}\, \delta_{X_k^*} + \frac{\alpha}{\alpha+n-1}\, P_0 \tag{1.1}$$

Poiché $X_1, \ldots, X_n$ sono scambiabili, questa distribuzione rimane valida per qualunque X_i condizionatamente a $X_1, \ldots, X_{i-1}, X_{i+1}, \ldots, X_n$.

L'esistenza di valori ripetuti induce anche una proprietà di *clustering*, dove i gruppi sono formati dalle osservazioni con uguale valore X_k^*. I distinti valori in $X_1, \ldots, X_n$, infatti, producono una partizione dell'insieme $\{1, \ldots, n\}$. Poiché le X_i sono casuali, il DP può essere visto come una distribuzione sulle partizioni di $\{1, \ldots, n\}$.

Dalla (1.1) si nota come ogni una nuova osservazione venga assegnata ad un gruppo già esistente con probabilità proporzionale alla sua numerosità: in questo modo i gruppi già più numerosi sono quelli che tendono a crescere ulteriormente. Il numero di *cluster*, infatti, risulta in genere molto minore del numero di osservazioni. Il numero di gruppi K_n atteso ha forma esplicita ed è pari a

$$\mathbb{E}[K_n] = \sum_{i=1}^{n} \frac{\alpha}{\alpha+i-1}$$

che, per $n \to \infty$ ha limite $\alpha \log \frac{n}{\alpha}$.

1.3.4 Rappresentazione *stick-breaking*

Una rappresentazione interessante del DP fu introdotta da Sethuraman (1994) ed è la cosiddetta rappresentazione *stick-breaking*. Si è già detto che le realizzazioni di un DP sono distribuzioni di probabilità discrete, quindi esprimibili come somme di punti di massa: questa caratteristica risulta evidente in questa costruzione del processo. Siano, per $k = 1, 2, \ldots$

$$\begin{gathered} X_k^* \overset{iid}{\sim} P_0 \qquad V_k \overset{iid}{\sim} Beta(1, \alpha) \\ \pi_1 = V_1 \qquad \pi_k = V_k \prod_{j=1}^{k-1} (1 - V_j) \;\; k \neq 1 \\ P = \sum_{k=1}^{\infty} \pi_k \delta_{X_k^*} \end{gathered} \tag{1.2}$$

allora $P \sim DP(\alpha, P_0)$. Il nome deriva da una metafora usata per descrivere la costruzione dei pesi: ogni V_k può essere visto come il frammento ottenuto spezzando infinite volte un bastoncino di lunghezza unitaria. Al primo passo il bastoncino viene spezzato ad una lunghezza V_1, pari alla massa assegnata al primo punto θ_1; la parte rimanente del bastoncino viene ulteriormente spezzata, ottenendo per il secondo punto un peso $\pi_2 = V_2(1 - V_1)$, e così via.

Questa rappresentazione è estremamente utile per più ragioni. Operando un troncamento opportuno, è possibile generare un DP in modo approssimato: ciò è utile nei casi in cui non è possibile ottenere la distribuzione a posteriori in forma analitica ed è necessario procedere per simulazione con metodi Monte Carlo. Inoltre, modificando la distribuzione beta della costruzione dello *stick-breaking*, è possibile ottenere nuove misure di probabilità. Un esempio è dato dal processo di Pitman-Yor, esposto di seguito.

1.4 Il processo di Pitman-Yor

Il processo di Pitman-Yor, $PY(\sigma, \theta, P_0)$, è una generalizzazione del processo di Dirichlet, che considera un ulteriore parametro $0 \leq \sigma < 1$. P_0 indica ancora la distribuzione di base e il parametro $\theta > -\sigma$ è equivalente al parametro di concentrazione del processo di Dirichlet, che può essere ottenuto come caso particolare ponendo $\sigma = 0$. Entrambe le rappresentazioni *stick-breaking* e delle urne di Pólya possono essere estese per descrivere tale processo.

In particolare, la rappresentazione *stick-breaking*, analoga alla (1.2), diventa la seguente: siano per $k = 1, 2, \ldots$

$$X_k^* \overset{iid}{\sim} P_0 \qquad V_k \overset{iid}{\sim} Beta(1-\sigma, \theta + k\sigma)$$
$$\pi_1 = V_1 \qquad \pi_k = V_k \prod_{j=1}^{k-1}(1 - V_j) \;\; k \neq 1$$
$$P = \sum_{k=1}^{\infty} \pi_k \delta_{X_i^*}$$

allora P si distribuisce come un processo $PY(\sigma, \theta, P_0)$.

La distribuzione predittiva invece diventa:

$$X_n | X_1, \ldots, X_{n-1} \sim \sum_{k=1}^{K} \frac{n_k - \sigma}{\theta + n - 1} \, \delta_{X_k^*} + \frac{\theta + K\sigma}{\theta + n - 1} \, P_0 \tag{1.3}$$

dove $(X_1^*, \ldots, X_K^*)$ sono ancora i K valori distinti in $X_1, \ldots, X_{n-1}$, con frequenze rispettivamente $(n_1, \ldots, n_K)$.

Dalla (1.3) risulta evidente il ruolo di σ: quando viene creato un nuovo *cluster*, $n_k = 1$. In questo caso, se $\sigma = 0$ (ovvero un DP), la probabilità che una seconda osservazione venga assegnata allo stesso gruppo è proporzionale a n_k, mentre, se $\sigma > 0$ tale probabilità risulta inferiore. Questo fa sì che inizialmente risulti molto difficile per un gruppo aumentare di numerosità. Alcune osservazioni cadranno comunque in un *cluster* già esistente: in questo caso, quando n_k aumenta, l'impatto di σ risulterà sempre più trascurabile e i gruppi già numerosi tenderanno a crescere ulteriormente. Ne risulteranno pochi *cluster* molto numerosi (come per il DP), e un gran numero di *cluster* con pochissime osservazioni.

Il numero atteso di gruppi distinti su n osservazioni, dato $P \sim PY(\sigma, \theta, P_0)$ è pari a:

$$\mathbb{E}[K_n] = \sum_{i=1}^{n} \frac{(\theta + \sigma)_{i+1}}{(\theta + 1)_{i+1}} \tag{1.4}$$

dove $(x)_n = \Gamma(x+n)/\Gamma(x) = x(x+1)\ldots(x+n-1)$.

1.5 Modelli mistura

Si consideri il problema della stima di una densità a partire da un campione di osservazioni indipendenti. Una stima parametrica richiederebbe di esplicitare la forma della distribuzione (ovvero la famiglia parametrica): ciò che non

è noto è solo il particolare elemento di tale famiglia, identificato univocamente da un parametro finito dimensionale. A volte, tuttavia, questo approccio può risultare troppo restrittivo oppure inadeguato rispetto alla conoscenza a disposizione sul processo che ha generato i dati. In questo caso, una stima nonparametrica permette di considerare completamente ignota la densità stessa: per rappresentare tale incertezza occorre specificare una distribuzione a priori adeguata, cioè una distribuzione sullo spazio delle funzioni di densità.

I processi esposti nella sezione precedente sono risultati molto utili a questo scopo: in particolare, di seguito è descritta la a priori che si ottiene come mistura di *kernel* parametrici dove la distribuzione misturante è un processo di Dirichlet. Utilizzare direttamente un DP come distribuzione a priori non è infatti la scelta più opportuna, poiché, sebbene le traiettorie siano misure di probabilità, queste sono discrete e quindi inadeguate per stimare una densità continua.

Siano dati uno spazio campionario $\mathcal{Y}$ e uno spazio parametrico Θ. Su $(\mathcal{Y}, \Theta)$ si consideri un *kernel* $\mathcal{K}(y, \theta)$, cioè una funzione tale che per ogni $\theta \in \Theta$, $\mathcal{K}(\cdot, \theta)$ è una densità su $\mathcal{Y}$. Se P è una distribuzione di probabilità su Θ; la densità marginale di Y è data da:

$$f_P(y) = \int \mathcal{K}(y, \theta) dP(\theta). \tag{1.5}$$

Su P si ponga una distribuzione a priori Π: questa, insieme al *kernel*, definisce una a priori sullo spazio $\mathcal{L}(\mathcal{Y})$ delle densità su $\mathcal{Y}$ attraverso la funzione $P \mapsto f_P(y)$. Se Π è un processo di Dirichlet, la distribuzione a priori sulla densità che si ottiene è detta *Dirichlet process mixture*, abbreviato DPM (Hjort et al., 2010; Müller et al., 2015).

Il modello ha un'utile rappresentazione gerarchica: siano $Y_1, \ldots, Y_n$ realizzazioni indipendenti, allora, per $i = 1, \ldots, n$:

$$\begin{aligned} Y_i|\theta_i &\sim \mathcal{K}(y_i, \theta_i) \\ \theta_i|P &\sim P \\ P &\sim DP(\alpha, P_0) \end{aligned} \tag{1.6}$$

Diversamente dai modelli parametrici, le quantità $\theta_1, \ldots, \theta_n$ non sono tutte uguali, ma assumono valori diversi tra i soggetti, in modo simile a degli effetti casuali, da una distribuzione P. Poiché P non è nota, se non la si vuole costringere ad appartenere ad una fissata famiglia parametrica, un processo di Dirichlet è una distribuzione a priori adeguata. Una conseguenza importante è che la distribuzione P sarà discreta: questo farà sì che tra i θ_i compaiano valori ripetuti, generando dei *cluster* all'interno del campione. Questo modello, infatti, oltre a permettere di stimare una densità

nonparametrica, è utile anche per identificare classi latenti che spiegano una dipendenza tra i soggetti.

Il modello può essere ottenuto come limite per $K \to \infty$ di una mistura finita con K componenti del tipo:

$$\begin{aligned} Y_i|C_i, \theta^* &\sim \mathcal{K}(y_i, \theta^*_{c_i}) \\ \theta^* &\sim P_0 \\ C_i|\pi &\sim Mult(\pi_1, \ldots, \pi_K) \\ \pi &\sim Dir(\alpha/K, \ldots, \alpha/K) \end{aligned} \tag{1.7}$$

Dove $(\theta^*_1, \ldots, \theta^*_K)$ sono i distinti valori in $(\theta_1, \ldots, \theta_n)$, e C è una variabile che indica la classe latente a cui appartiene ciascuna osservazione. I valori che può assumere la variabile C sono arbitrari, poiché sono solo delle "etichette" che identificano i gruppi.

La scelta del *kernel* dipende dalla natura dei dati: se $\mathcal{Y}$ è l'asse reale, un *kernel* normale è adatto, se invece $\mathcal{Y} = \mathbb{R}^+$, una mistura di gamma o di Weibull risulta più adeguata.

Il modello può inoltre essere esteso considerando un processo stocastico diverso da quello di Dirichlet, come, per esempio, il processo di Pitman-Yor.

1.5.1 Consistenza dell'a posteriori

Si considerino $(Y_1, \ldots, Y_n)$ indipendenti e identicamente distribuite con densità p_θ, condizionatamente a $\theta \in \Theta$. Sia Π una distribuzione a priori, ovvero una distribuzione di probabilità su Θ e sia

$$\Pi(B|Y_1, \ldots, Y_n) = \frac{\int_B p_\theta(y_1, \ldots, y_n) d\Pi(\theta)}{\int_\Theta p_\theta(y_1, \ldots, y_n) d\Pi(\theta)}$$

la distribuzione a posteriori sul sottoinsieme $B \subset \Theta$.

Si assuma che esista un vero valore θ_0 per il parametro, a cui corrisponde una densità p_{θ_0}. La distribuzione a posteriori, $\Pi(\cdot|Y_1, \ldots, Y_n)$, è detta consistente a θ_0 se per ogni intorno U di θ_0,

$$\Pi(U|Y_1, \ldots, Y_n) \to 1$$

per $n \to \infty$, in probabilità o quasi certamente.

Questa proprietà indica se la distribuzione a posteriori, al crescere dell'informazione campionaria, si concentra con probabilità 1 in un intorno del vero parametro, cioè se sarebbe in grado di identificare il vero processo che ha generato i dati se si avesse a disposizione un numero infinito di osservazioni. Anche se nella realtà non è una situazione realizzabile, una a posteriori

non consistente è indicazione che l'inferenza può essere scorretta anche con campioni finiti.

Nel caso parametrico la consistenza è quasi sempre garantita, infatti la a posteriori è consistente ovunque, tranne sui sottoinsiemi a cui la a priori assegna probabilità nulla. È sufficiente, quindi, scegliere la a priori in modo che sia sufficientemente ampia e non escluda alcuni valori di Θ. Nel caso nonparametrico, tuttavia, quando lo spazio Θ ha dimensione infinita, la condizione di assegnare probabilità a priori non nulla agli intorni del vero parametro non è sufficiente a garantire la consistenza.

Il caso nonparametrico con $\Theta = \mathcal{L}$, spazio delle densità, è particolarmente rilevante: in un teorema, Schwartz (1965) definisce delle condizioni sul modello e sul supporto della a priori che garantiscono la consistenza della posteriori, quando la topologia di riferimento è la topologia debole. Si consideri quindi questo spazio, su cui è definita una distribuzione a priori Π, e sia $f_0 \in \mathcal{L}$ la vera densità che ha generato i dati. La prima condizione riguarda il supporto della distribuzione a priori: in particolare, si richiede che la a priori assegni probabilità positiva a ogni intorno di f_0, dove l'intorno tuttavia non è definito secondo la topologia di interesse, ma rispetto alla divergenza di Kullback-Leibler. Questa proprietà può anche essere espressa dicendo che f_0 deve appartenere al supporto di Kullback-Leibler della a priori, ovvero $\Pi(K_\epsilon(f_0)) > 0$, dove $K_\epsilon(f_0) = \{f : KL(f_0, f) < \epsilon;\ \epsilon > 0\}$. La seconda condizione richiede che f_0 e U^c, dove U è un intorno di f_0, siano separabili. Questa idea è formalizzata attraverso l'esistenza di una sequenza di funzioni test uniformemente consistenti (cioè tali per cui entrambi gli errori di I e II tipo convergono a zero esponenzialmente) per la verifica di ipotesi $H_0 : f = f_0$ contro $H_1 : f \in U^c$. Nel caso la topologia di riferimento sia la topologia debole, questo tipo di test esiste sempre, quindi la condizione sul supporto di Kullback-Leibler della a priori è sufficiente a garantire la consistenza debole. Se, tuttavia, si considera una topologia più forte per U, come per esempio quella indotta dalla metrica L_1, è necessario introdurre delle condizioni ulteriori, poiché in questo caso un test uniformemente consistente non esiste. Senza entrare nel dettaglio, si richiedono delle condizioni sulla dimensione dello spazio parametrico, misurata rispetto all'entropia in metrica L_1. Per un generico sottoinsieme $\mathcal{G} \subset \mathcal{L}$, questa è definita come il logaritmo del più piccolo k tale per cui esistono $f_1, \ldots, f_k$ tali che $\mathcal{G} \subset \bigcup_{j=1}^{k} \{f : \|f - f_j\| < \delta\}$, $\delta > 0$, ed è indicata con $\mathcal{J}(\delta, \mathcal{G})$. Per stabilire la consistenza forte, è necessario poter definire una sequenza $\mathcal{F}_n$ di sottoinsiemi di $\mathcal{L}$, indicizzati da n, che crescono a coprire $\mathcal{L}$: per n grande, questi devono essere tali per cui la probabilità che Π assegna a $\mathcal{F}_n^c$ deve essere esponenzialmente piccola; mentre $\mathcal{J}(\delta, \mathcal{F}_n) < n\beta$ per un'opportuno $\beta > 0$ (Ghosal et al., 1999; Hjort et al., 2010).

Queste condizioni possono essere applicate alla distribuzione a priori sullo spazio delle densità indotta da un DPM per verificare la consistenza della posteriori che ne deriva. Per esempio, se il *kernel* scelto è la densità normale di media μ e varianza σ^2, $N(\mu, \sigma)$, si può verificare che la densità che si ottiene come $f_P(y) = \int N(y; \mu, \sigma) dP(\mu, \sigma)$ converge a qualunque densità per $\sigma \to 0$ rispetto alla metrica L_1.

1.6 Metodi MCMC per l'inferenza sulla posteriori

L'inferenza per i modelli basati sul processo di Dirichlet o di Pitman-Yor non è possibile in via analitica, ma esistono diversi metodi di tipo *Markov Chain Monte Carlo* per la simulazione dalla distribuzione a posteriori, in particolare basati sul Gibbs sampler. Nel caso del DPM, esistono due "classi" di algoritmi: marginali oppure condizionali.

I primi sono detti marginali poiché la distribuzione di probabilità casuale P è integrata via dal modello, in modo da eliminare il problema di simulare da una misura infinito-dimensionale: in questo caso si sfrutta la rappresentazione delle urne di Pólya del processo. I secondi, invece, si basano sulla costruzione *stick-breaking*.

Di seguito sono descritti quattro algoritmi: i primi due fanno parte dei metodi marginali, mentre gli ultimi di quelli condizionali.

1.6.1 Pólya urn Gibbs sampler

Si consideri il modello che si ottiene dalla (1.6) integrando via P: il metodo più diretto per ottenere un campione dalla a posteriori è estrarre ogni θ_i dalla distribuzione condizionata ai dati e a θ_j per $j \neq i$, in seguito indicato con θ_{-i}. Questa distribuzione si può ottenere combinando la verosimiglianza di θ_i, cioè $\mathcal{K}(y_i, \theta_i)$, con la distribuzione a priori di θ_i condizionata a θ_{-i}.

Un particolare Gibbs sampler si ottiene nel caso in cui la distribuzione di $\theta_i|\theta_{-i}$ sia esprimibile come urne di Pólya (Neal, 2000): nel caso si utilizzi un processo di Dirichlet $DP(\alpha, P_0)$, per esempio, questa è pari a

$$\theta_i|\theta_{-i} \sim \frac{1}{\alpha + n - i} \sum_{j \neq i} \delta(\theta_j) + \frac{\alpha}{\alpha + n - 1} P_0$$

Data questa forma per la a priori e la verosimiglianza $\mathcal{K}(y_i, \theta_i)$, si ricava da $p(\theta_i|\theta_{-i}, y_i) \propto p(\theta_i|\theta_{-i})\mathcal{K}(y_i, \theta_i)$, per $i = 1, \ldots, n$, la seguente distribuzione

$$\theta_i|\theta_{-i}, Y_i \sim \sum_{j \neq i} q_{i,j}\delta(\theta_j) + r_i H_i \tag{1.8}$$

con

$$q_{i,j} = b\,\mathcal{K}(y_i, \theta_j)$$
$$r_i = b\,\alpha \int \mathcal{K}(y_i, \theta) dP_0(\theta)$$

Dove b è tale per cui $\sum_{j \neq i} q_{i,j} + r_i = 1$.
La distribuzione H_i corrisponde alla distribuzione a posteriori di θ data l'osservazione y_i, cioè:

$$H(\theta_i|y_i) = \frac{\mathcal{K}(y_i, \theta_i)P_0(\theta_i)}{\int \mathcal{K}(y_i, \theta) dP_0(\theta)}$$

Se i valori correnti della catena sono $\theta = (\theta_1, \ldots, \theta_n)$, a ogni passo successivo si estrae, per $i = 1, \ldots, n$, un nuovo valore θ_i dalla (1.8); ovvero:

$$\begin{cases} \theta_i = \theta_k^* & \text{con probabilità } \propto n_{-i,k}\,\mathcal{K}(y_i, \theta_k^*) \quad k = 1, \ldots, K_{-i} \\ \theta_i \sim H_i & \text{con probabilità } \propto \alpha \int \mathcal{K}(y_i, \theta) dP_0(\theta) \end{cases}$$

Questo algoritmo può essere facilmente implementato se le operazioni di calcolo dell'integrale e simulazione dalla distribuzione H non risultano troppo complicate: il caso ottimale si ha quando P_0 è coniugata a $\mathcal{K}$.

Questo metodo è tuttavia molto inefficiente, nel senso che la catena impiega molti passi per raggiungere la convergenza. Il problema nasce dalla presenza di gruppi di osservazioni che hanno, con probabilità alta, lo stesso valore di θ (ovvero alto $q_{i,j}$). In questo caso, infatti, un cambiamento nel valore di θ avverrà solo raramente: poiché l'algoritmo permette di aggiornare un solo parametro alla volta, necessariamente si dovrà passare per una fase intermedia in cui le osservazioni dello stesso *cluster* vengono associate a parametri diversi (e ciò accade con probabilità bassa).

Questo problema può essere evitato considerando il modello nella forma (1.7) con la proporzione di mistura π integrata via. La presenza della variabile aggiuntiva C permette di costruire un Gibbs sampler che a ogni passo genera prima l'appartenenza al *cluster* per ogni osservazione, $c = (c_1, \ldots, c_n)$, e poi, condizionatamente a questa, aggiorna il parametro θ_c^* corrispondente. In questo modo il valore del parametro associato a un gruppo cambia contemporaneamente per tutte le osservazioni che ne fanno parte.

1.6.2 Algoritmo "*no gaps*"

Un particolare algoritmo, introdotto da MacEachern e Müller (1998) e in seguito ripreso da Neal (2000), si ottiene vincolando i valori $(c_1, \ldots, c_n)$ ad appartenere alla sequenza degli interi da 1 a k, con k numero di distinti c_i, ed è per questo detto "*no gaps*". Questo metodo fu introdotto come soluzione nei casi in cui P_0 e $\mathcal{K}$ non sono coniugate: l'aggiornamento di c, infatti, non richiede di dover simulare da H_i e di calcolare l'integrale $\int \mathcal{K}(y_i, \theta) dP_0(\theta)$. Per fare ciò, vengono introdotti m parametri ausiliari, che corrispondono ai potenziali parametri dei futuri *cluster*, non ancora occupati. A ogni passo dell'algoritmo si avranno, oltre ai valori correntemente associati a qualche osservazione $\theta^* = \{\theta_1^*, \ldots, \theta_k^*\}$, m ulteriori parametri temporanei $\theta_E^* = \{\theta_{k+1}^*, \ldots, \theta_{k+m+1}^*\}$, estratti indipendentemente da P_0.

In particolare, esposta di seguito è la versione proposta da Neal, nel caso $m = 1$. Siano $c = (c_1, \ldots, c_n)$ e $\theta^* = (\theta_c^* : c \in \{c_1, \ldots, c_n\})$ i valori della catena; l'algoritmo consiste nei seguenti passi:

(a) per $i = 1, \ldots, n$: sia c_{-i} l'insieme c senza l'i-esima componente, con k_{-i} valori distinti; si numerino quindi i c_{-i} con i valori $\{1, \ldots, k_{-i}\}$. Si estragga un nuovo $\theta^*_{k_{-i}+1} \sim P_0$ per la variabile ausiliaria. Un nuovo valore c_i si ottiene quindi dalla seguente distribuzione:

$$\mathbb{P}[c_i = c | c_{-i}, y_i, \theta_1^*, \ldots, \theta^*_{k_{-i}+1}] \propto \begin{cases} \frac{n_{-i,c}}{\alpha+n-1} \mathcal{K}(y_i, \theta_c^*) & \text{se } c \in \{1, \ldots, k_{-i}\} \\ \frac{\alpha}{\alpha+n-1} \mathcal{K}(y_i, \theta^*_{k_{-i}+1}) & \text{se } c = k_{-i} + 1 \end{cases}$$

dove $n_{-i,c}$ è il numero di $c_j = c$ in c_{-i}. Si aggiorni il vettore θ^* in modo che contenga solo i valori correntemente associati ad un *cluster*.

(b) per ogni $c \in \{c_1, \ldots, c_n\}$: si aggiorna θ^* estraendo un nuovo valore dalla distribuzione a posteriori $\theta_c^* | y$.

1.6.3 Blocked Gibbs sampler

Questo algoritmo fa parte dei metodi condizionali, e può essere applicato nei modelli bayesiani nonparametrici in cui la distribuzione a priori ammette una costruzione di tipo *stick-breaking*. Evitare di marginalizzare la a priori comporta alcuni vantaggi: la convergenza risulta spesso più veloce e l'inferenza sulla posteriori di P è semplificata e formalmente più corretta.

In generale, una a priori *stick-breaking* è una misura di probabilità casuale discreta, che ammette una forma del tipo $P_N = \sum_{k=1}^N \pi_k \delta_{\theta_k^*}$, con $0 \leq \pi_k \leq 1$ e $\sum_{k=1}^N \pi_k = 1$. Ciò che caratterizza queste distribuzioni è la particolare costruzione dei pesi per mezzo di variabili beta indipendenti.

Il blocked Gibbs sampler (Ishwaran e James, 2001) si basa sull'assunzione che la a priori sia di dimensione finita, ovvero che $N < +\infty$. I processi di Dirichlet o di Pitman-Yor non soddisfano questa ipotesi, per questo è necessario un opportuno troncamento a un numero N finito di componenti. Questo si ottiene scartando i successivi $N+1, N+2, \ldots$ termini della misura infinito-dimensionale P_∞, e ponendo $\pi_N = 1 - \pi_1 - \cdots - \pi_{N-1}$.

Il livello di troncamento N deve essere scelto adeguatamente: nel caso dei modelli esposti nella sezione 1.5, la densità marginale che risulta utilizzando come a priori P_N deve essere indistinguibile dal limite che si otterrebbe con P_∞. Indicando con μ questa densità, Ishwaran e James (2001) derivano un limite esplicito per la distanza $\|\mu_N - \mu_\infty\|_1$, dove $\|\cdot\|_1$ indica la distanza L_1. Nel caso P_∞ sia il processo di Pitman-Yor $PY(\theta, \sigma, P_0)$, si ha:

$$\|\mu_N - \mu_\infty\|_1 \leq 4(1 - \mathbb{E}[1 - T_N(1, \theta, \sigma)]^n)$$

dove $T_N(r, \theta, \sigma) = (\sum_{k=N}^{\infty} \pi_k)^r$ con $\mathbb{E}[T_N(r, \theta, \sigma)] = \prod_{k=1}^{N-1} \frac{(\theta + k\,\sigma)_r}{(\theta + (k-1)\,\sigma + 1)_r}$. Mentre, nel caso di un processo di Dirichlet $DP(\alpha, P_0)$ si ha:

$$\|\mu_N - \mu_\infty\|_1 \sim 4ne^{-\frac{N-1}{\alpha}}.$$

Per il processo di Dirichlet tale distanza diminuisce esponenzialmente con N, quindi è sufficiente un numero limitato di componenti per approssimare adeguatamente μ_∞. Nel caso del processo di Pitman-Yor, invece, per alcune combinazioni di parametri è necessario un numero N molto grande per raggiungere la stessa accuratezza. Questo argomento verrà ripreso anche nel capitolo successivo, in cui, attraverso la simulazione, si indagherà meglio il problema del numero di componenti necessarie per rappresentare adeguatamente il processo di Pitman-Yor.

1.6.4 Slice Gibbs sampler

Anziché operare un troncamento di P, che produce comunque un'approssimazione del processo, un approccio alternativo si basa sull'introduzione di una variabile latente, condizionatamente alla quale P è finita. Walker (2007) e Kalli et al. (2011) propongono un algoritmo basato sullo *slice sampling*.

Questo Gibbs sampler può essere applicato a tutti i modelli che generano una densità casuale del tipo (1.5), dove la a priori ammette una rappresentazione *stick-breaking* $P = \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k^*}$: data questa forma, infatti, la densità può essere scritta come mistura discreta infinita:

$$f_{\pi, \theta^*}(y) = \sum_{k=1}^{\infty} \pi_k \mathcal{K}(y, \theta_k^*)$$

L'idea di questo metodo è di introdurre una variabile latente u che abbia densità congiunta

$$f_{\pi,\theta^*}(y,u) = \sum_{k=1}^{\infty} I(u < \pi_k)\mathcal{K}(y,\theta_k^*) \tag{1.9}$$

tale che la densità marginale di y che si ottiene integrando u sia $f_{\pi,\theta^*}(y)$. La (1.9) può anche essere scritta come

$$f_{\pi,\theta^*}(y,u) = \sum_{k=1}^{\infty} \pi_k U(u;0,\pi_k)\mathcal{K}(y,\theta_k^*)$$

dove $U(\cdot;a,b)$ indica la densità di una variabile uniforme sull'intervallo $[a,b]$. Una conseguenza importante è che, condizionatamente a u, la densità di y può essere espressa per mezzo di un numero finito di componenti, infatti

$$f_{\pi,\theta^*}(y|u) = \frac{1}{f_\pi(u)} \sum_{k:\pi_k > u} \mathcal{K}(y,\theta_k^*)$$

dove $f_\pi(u) = \sum_{k=1}^{\infty} I(u < \pi_k)$ è la densità marginale di u, ed è definita su $[0, \max_k \pi_k]$.

L'insieme delle componenti effettivamente utilizzate è $A_k(u) = \{k : \pi_k > u\}$: questo insieme, per $u > 0$ fissato, è sicuramente finito. Il modello può, quindi, essere riscritto come:

$$f_{\pi,\theta^*}(y|u) = \frac{1}{\sum_{k\in A_k(u)} \pi_k} \sum_{k\in A_k(u)} \mathcal{K}(y,\theta_k^*).$$

Condizionatamente a u, il modello che ne risulta è una mistura finita, con pesi uguali e pari a $1/\sum_{k\in A_k(u)} \pi_k$. Anziché operare un troncamento deterministico di P, come nel blocked Gibbs sampler, questo algoritmo opera un troncamento "dinamico", che a ogni passo definisce diversamente le componenti da tenere e quelle da scartare per mezzo di una variabile ausiliaria.

Un ultimo passo consiste nell'introdurre un'ulteriore variabile latente che identifica la componente della mistura a cui y appartiene. Indicando con c questa variabile, si ottiene la densità congiunta:

$$f_{\pi,\theta^*}(y,c,u) = I(c \in A_\pi(u))\ \mathcal{K}(y,\theta_c^*).$$

Per un campione di osservazioni di numerosità n, la verosimiglianza si ottiene facilmente come prodotto di queste densità.

Si consideri il modello appena descritto, in cui P è un processo di Dirichlet $DP(\alpha, P_0)$; l'algoritmo consiste nei seguenti passi:

(a) aggiornamento di u: per $i = 1, \ldots, n$ si estrae

$$u_i \sim Unif(0, \pi_{c_i})$$

(b) aggiornamento di θ^*:

$$f(\theta_k^*|-) \propto P_0(\theta_k^*) \prod_{i:c_i=k} \mathcal{K}(y_i, \theta_k^*).$$

Se nessun $c_i = k$, allora $f(\theta_k^*|-) \propto P_0(\theta_k^*)$.

(c) aggiornamento delle variabili v_k per la costruzione dei pesi casuali π:

$$f(v_k|-) \propto Beta(1 + \sum_{i=1}^{n} I(c_i = k), \alpha + \sum_{i=1}^{n} I(c_i > k)).$$

Da questo passo risulta evidente come l'algoritmo possa essere applicato anche a varianti del modello che considerano priori diverse dal DP, purché ammettano una costruzione di tipo *stick-breaking*. Se, in generale, si considera $V_k \sim Beta(a_k, b_k)$, l'aggiornamento dei parametri è $a'_k = a_k + \sum_{i=1}^{n} I(c_i = k)$, $b'_k = b_k + \sum_{i=1}^{n} I(c_i > k)$.

(d) aggiornamento della variabile indicatrice c:

$$\mathbb{P}[C_i = k|-] \propto I(k \in A_\pi(u_i))\mathcal{K}(y_i, \theta_k^*)$$

I possibili valori che può assumere questa variabile sono limitati, infatti ogni c_i può assumere solo i valori $\{k : \pi_k > u_i\}$. Senza la variabile ausiliaria u, l'indicatrice potrebbe invece assumere un'infinità di valori.

1.7 Modelli per dati di conteggio

Come si vedrà nel Capitolo 3, i dati di interesse sono costituiti da conteggi. Nel caso la variabile Y di cui si vuole stimare la distribuzione non sia continua ma discreta, in particolare a valori sull'insieme dei numeri naturali, il modello parametrico più semplice e intuitivo è il modello Poisson. Questo modello è, tuttavia, poco flessibile, a causa dell'unico parametro λ che controlla sia la media che la varianza, assunte uguali e pari a λ: questa ipotesi in genere non è verificata e ciò rende la distribuzione inadeguata nella maggior parte dei casi.

Una situazione frequente nei dati reali è che essi presentino, invece, sovradispersione, ovvero varianza maggiore di quella attesa. Per risolvere questo

problema, un particolare modello si ottiene ipotizzando che Y abbia distribuzione Poisson di media λ, dove, tuttavia, λ è anch'essa una variabile aleatoria, a valori in $\mathbb{R}^+$: ciò equivale a porre un effetto casuale sulla media. La variabile λ non è osservabile, infatti, ciò che si osserva è solo la distribuzione marginale di Y,

$$\mathbb{P}[Y = j|P] = \int Poi(j, \lambda) dP(\lambda) \tag{1.10}$$

dove P è la distribuzione di λ: questo modello definisce una mistura di *kernel* Poisson. Una scelta tipica per P è la distribuzione gamma: questa induce per la marginale di Y una distribuzione binomiale negativa (Hougaard et al., 1997).

In alcuni casi rilassare solo l'ipotesi sulla varianza della distribuzione non è sufficiente: in questi casi è necessario un approccio più flessibile. Una possibilità è utilizzare una mistura Poisson nonparametrica, scegliendo per P nella (1.10), anziché una fissata distribuzione, un processo di Dirichlet, in modo analogo ai modelli visti nella sezione 1.5 per variabili continue. Questo modello è all'apparenza molto flessibile, tuttavia, i problemi della distribuzione Poisson rispetto alla sovradispersione o sottodispersione persistono. Tutte le distribuzioni che presentano sottodispersione, per esempio, non sono contenute nel supporto di questo modello, e non potranno essere stimate in modo consistente (Canale e Dunson, 2011; Canale e Prünster, 2017).

Un'alternativa più flessibile si può ottenere considerando, anziché una mistura di Poisson, una mistura di multinomiali: questo modello, tuttavia, oltre a richiedere un limite per il supporto, è anche troppo flessibile, poiché parametrizzato da un numero di parametri pari al numero di distinti valori del supporto.

Analogamente al caso della stima di una densità continua, ciò che si vorrebbe ottenere nel caso discreto è un modello che sia abbastanza flessibile da permettere di approssimare un gran numero di distribuzioni, ma che allo stesso tempo abbia un numero di parametri limitato. A questo scopo, Canale e Dunson (2011) propongono una classe di misture nonparametriche per dati di conteggio basate su particolari *kernel*, ottenuti attraverso arrotondamento di *kernel* continui. L'idea è di considerare una variabile continua sottostante con densità f: una distribuzione a priori per la distribuzione di probabilità discreta p sarà quindi definita a partire dalla a priori su f.

In particolare, sia $Y \in \mathbb{N}$ una variabile discreta con distribuzione p, e $Y^* \in \mathcal{Y}$ una variabile continua con densità f. Le due variabili sono legate da una relazione $Y = h(Y^*)$, dove $h(\cdot)$ è una funzione tale che $h(Y^*) = j$ se $Y^* \in (a_j, a_{j+1}]$. La distribuzione p può essere ottenuta a partire da f,

$p = g(f)$, grazie a una funzione $g(\cdot)$ tale che:

$$p(j) = g(f)[j] = \int_{a_j}^{a_{j+1}} f(y^*)dy^* \qquad j \in \mathbb{N} \tag{1.11}$$

dove le $\{a_j\}_{j=0,1,\ldots,\infty}$ sono delle soglie, fissate, su $\mathcal{Y}$, con $a_0 = \min\{y^* : y^* \in \mathcal{Y}\}$ e $a_\infty = \max\{y^* : y^* \in \mathcal{Y}\}$.
Una scelta conveniente per f è il *kernel* normale $N(\mu, \tau^{-1})$, con τ parametro di precisione, e soglie pari a $a_0 = -\infty$, $a_j = j$ per $j = 1, \ldots, +\infty$: la distribuzione discreta che ne risulta è in seguito indicata con $RG(\mu, \tau^{-1})$ (*Rounded Gaussian*).

Al fine di definire una distribuzione a priori sullo spazio $\mathcal{C}$ delle distribuzioni di probabilità discrete, si consideri pertanto la a priori Π^* sullo spazio $\mathcal{L}$ delle densità che si ottiene come:

$$f_P(y^*) = \int N(y^*; \mu, \tau^{-1})dP(\mu, \tau)$$
$$P \sim \tilde{\Pi}$$

dove $\tilde{\Pi}$ indica la a priori su P, per esempio un processo di Dirichlet o di Pitman-Yor. Una distribuzione a priori Π su $\mathcal{C}$ è quindi definita grazie alla relazione (1.11) a partire dalla distribuzione a priori Π^* sulla densità sottostante.

La distribuzione a posteriori $\Pi(\cdot|Y_1, \ldots, Y_n)$ che ne deriva gode di alcune buone proprietà. La prima riguarda la consistenza debole: dalla teoria di Schwartz si è visto come questa sia legata alla proprietà di Kullback-Leibler della distribuzione a priori. Si può dimostrare che la funzione $g : \mathcal{L} \to \mathcal{C}$ mantiene gli intorni di KL, nel senso che, se f_0 è una densità tale che $p_0 = g(f_0)$ e $K_\epsilon(f_0)$ è un intorno di KL di f_0, allora $g(K_\epsilon(f_0))$ contiene un intorno di KL di p_0. La posteriori che si ottiene a partire da Π, quindi, è debolmente consistente a ogni $p_0 \in \mathcal{C}$ se almeno un elemento dell'insieme $g^{-1}(p_0)$ appartiene al supporto di KL di Π^*.

La seconda proprietà riguarda, invece, la consistenza forte: nel caso di una a priori sullo spazio delle distribuzioni discrete, la consistenza debole della posteriori implica la consistenza forte, rispetto alla metrica L_1.

Capitolo 2

Studi di simulazione

Al fine di applicare i metodi appena descritti a un insieme di dati reali, come si vedrà nel capitolo successivo, sono state condotte alcune simulazioni per indagare i risultati degli algoritmi e dei modelli esposti. In particolare, gli aspetti che verranno studiati sono due: da un lato, l'adeguatezza di diverse specificazioni della distribuzione a priori, dall'altro, il funzionamento dello slice Gibbs sampler e del Pólya urn Gibbs sampler nei casi considerati.

I dati che verranno presi in esame, descritti più nel dettaglio in seguito, sono costituiti dai conteggi del numero di morti all'interno di gruppi di individui, per un totale di 72 gruppi. L'obiettivo è stimarne la distribuzione di probabilità usando misture nonparametriche per dati discreti del tipo descritto nella Sezione 1.7.

Attraverso la simulazione si vogliono confrontare i risultati che si ottengono a partire da una mistura di *kernel* Poisson e da una più flessibile mistura di *kernel* RG. Per semplicità, in entrambi i modelli si sono scelte delle distribuzioni di base che fossero coniugate: nel caso della mistura Poisson si è posta, quindi, una distribuzione $Gamma(\alpha, \beta)$, di parametri $\alpha = 2$, $\beta = 2$; per la mistura di RG, parametrizzata dal parametro di precisione, una distribuzione normale-gamma $N(\mu; \mu_0, \kappa\tau^{-1})Gamma(\tau; \alpha, \beta)$. I parametri μ_0 e κ sono stati fissati pari, rispettivamente, alla media e alla varianza campionarie, mentre i parametri α e β sono stati posti pari a 4 e a 2, rispettivamente.

Un secondo punto importante è la scelta del processo stocastico utilizzato come distribuzione misturante. Se si utilizza un DP, infatti, il numero di componenti stimato a posteriori tende ad essere fortemente influenzato dalla specificazione del parametro, ovvero dal numero di gruppi atteso a priori. Per questo motivo verrà usato un processo di Pitman-Yor che, all'aumentare di σ, fornisce risultati più robusti rispetto alla specificazione a priori (Canale e Prünster, 2017). Per verificare via simulazione questa proprietà, il parametro σ verrà fatto variare in un range di valori a partire da 0 (che corrisponde a

utilizzare un DP) fino a 0.50; mentre θ sarà calcolato grazie alla (1.4) in modo da avere, a priori, un numero di gruppi atteso pari a $5, 10, 15, 20$.

Sono stati simulati due diversi scenari: una mistura di *kernel* RG e una di *kernel* Poisson, in modo che per entrambi i modelli d'interesse fosse possibile verificare l'adattamento sia nel caso si tratti del vero processo generatore dei dati, sia nel caso di un'errata specificazione. Per ciascuno di questi scenari si sono considerate due diverse specificazioni del numero di componenti, $k_0 = 3, 6$, e tre diverse numerosità campionarie, $n = 50, 70, 100$. Per ogni scenario sono state simulate catene di lunghezza 7000, con un *burn-in* di 3000. A causa del grande numero di catene simulate, non sono stati condotti test formali per valutarne la convergenza (come invece verrà fatto nel caso dell'applicazione su dati reali nel capitolo successivo), tuttavia, l'analisi grafica non evidenziava comportamenti problematici.

2.1 Confronto tra algoritmi

Il primo algoritmo scelto è lo slice Gibbs sampler, poiché, come si è visto, comporta dei vantaggi sia rispetto alla classe dei metodi marginali, sia rispetto al blocked Gibbs sampler. In particolare, quest'ultimo non sarebbe stato applicabile a causa della scelta di usare un processo di Pitman-Yor, che avrebbe richiesto un livello di troncamento eccessivamente elevato.

Per implementare lo slice Gibbs sampler in un programma è comunque necessario definire il numero massimo di componenti a disposizione per rappresentare nel calcolatore il processo, cioè la lunghezza del vettore dei pesi casuali π, generati tramite *stick-breaking*: in questo caso si è scelto un limite di 500. Per controllare il numero di componenti effettivamente utilizzate dall'algoritmo, si è tenuto traccia per ogni catena del massimo valore assunto dalla variabile che identifica la componente di mistura, indicata con c nella sezione 1.6.4.

In Tabella A.1 e A.2 sono riportati i risultati dell'applicazione di una mistura di *kernel* Poisson ai due scenari descritti in precedenza, mentre in Tabella A.3 e A.4 sono riportati i risultati della mistura di *kernel* RG. In ogni tabella sono indicate la numerosità campionaria del data set n, il vero numero di gruppi da cui sono simulati i dati, indicato con k_0, e il numero di gruppi atteso a priori $\mathbb{E}[K_n]$. Le colonne successive riportano, per i valori di σ considerati, la media a posteriori del numero di gruppi stimato e il massimo numero di componenti utilizzate dall'algoritmo, indicato con max_c.

Un fenomeno evidente in tutti i casi considerati riguarda il numero di componenti utilizzate: mentre nel caso di un DP ($\sigma = 0$) il valore max_c è contenuto (spesso poche decine) anche per numerosità campionarie eleva-

te, già per $\sigma = 0.25$ è spesso oltre 400, per diventare quasi sempre pari al limite di 500 nel caso $\sigma = 0.50$. Per rappresentare adeguatamente il processo sarebbe quindi necessario definire un numero maggiore di componenti e di conseguenza anche matrici di dimensione superiore. Tuttavia, se già nel caso in cui la numerosità campionaria è pari a 50 l'algoritmo usa tutte le componenti a disposizione, il rischio è che per numerosità superiori le dimensioni delle matrici diventerebbero proibitive e comporterebbero un'eccessivo, e inutile, utilizzo di RAM. Si è scelto quindi di abbandonare lo slice Gibbs sampler in favore di un algoritmo marginale. Questi algoritmi, oltre a non operare troncamenti del processo, necessitano di matrici di dimensione al più pari alla numerosità campionaria.

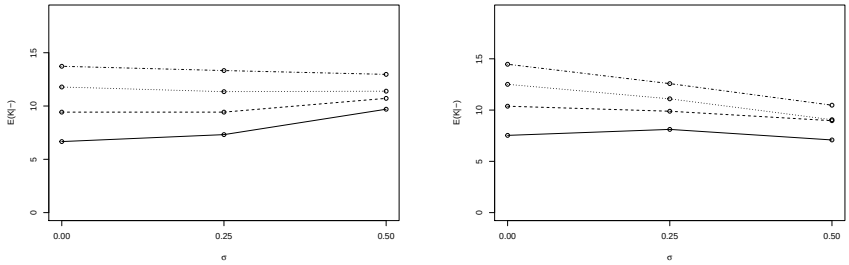
2.2 Confronto tra modelli

Visti i risultati delle simulazioni con i due diversi algoritmi, per studiare i risultati dei modelli di interesse, ovvero le misture di *kernel* Poisson e RG, sono state utilizzate le catene ottenute tramite Pólya urn Gibbs sampler, in particolare con l'algoritmo "no gaps" descritto nella Sezione 1.6.2.

In Tabella A.5 sono riportate le medie a posteriori del numero di gruppi stimato con la mistura di *kernel* Poisson, mentre in Tabella A.6 le stime ottenute tramite mistura di *kernel* RG.

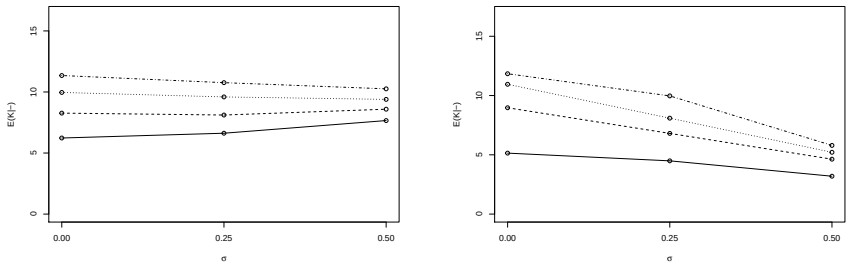
Si consideri il caso della mistura di *kernel* Poisson: quasi sempre il numero di gruppi è sovrastimato, tuttavia rimane comunque intorno a valori ragionevoli. Sugli stessi dati, per i diversi valori di θ considerati (cioè diverse specificazioni del numero di gruppi atteso a priori), all'aumentare di σ non si nota una convergenza netta ad un valore unico. Tuttavia, per le stime associate ai θ inferiori, cioè $\mathbb{E}[K_n]$ pari a 5 o 10, all'aumentare di σ si nota come queste tendano sempre ad aumentare e si avvicinino a quelle che si ottengono per θ elevato. Considerare anche una specificazione del processo con σ superiore a 0.50 avrebbe permesso di evidenziare meglio questo fenomeno, tuttavia, le catene tendevano a diventare instabili con conseguenti problemi sulla convergenza. Questo comportamento del *kernel* Poisson è comunque coerente con quanto mostrato più nel dettaglio da Canale e Prünster (2017).

Per quanto riguarda le stime del numero di componenti ottenute assumendo una mistura di *kernel* RG, si nota come la convergenza all'aumentare di σ risulti più netta e, soprattutto, ad un valore più corretto rispetto al caso Poisson. Anche in questo caso il modello tende a sovrastimare leggermente il numero di gruppi, tuttavia, questo comportamento non è preoccupante: è noto, infatti, come sia il DP sia il PYP tendano a sovrastimare il numero di *cluster* presenti nei dati, poiché a ogni iterazione dell'algoritmo sono in gene-



(a) Mistura di Poisson (b) Mistura di RG

Figura 2.1: Pólya urn Gibbs sampler: media a posteriori del numero di componenti. Dati simulati da una mistura di 3 Poisson.



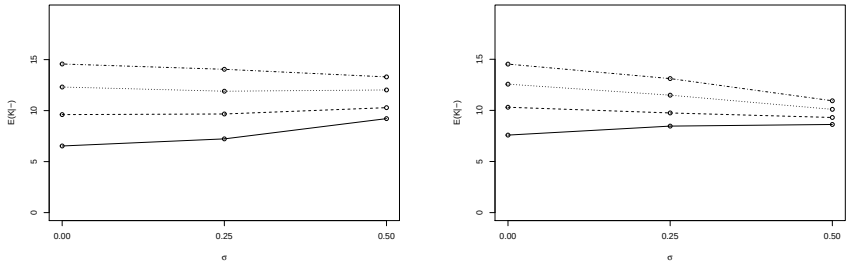
(a) Mistura di Poisson (b) Mistura di RG

Figura 2.2: Pólya urn Gibbs sampler: media a posteriori del numero di componenti. Dati simulati da una mistura di 6 Poisson.

re presenti dei piccoli *cluster* transitori composti da pochissime osservazioni (Miller e Harrison, 2014).

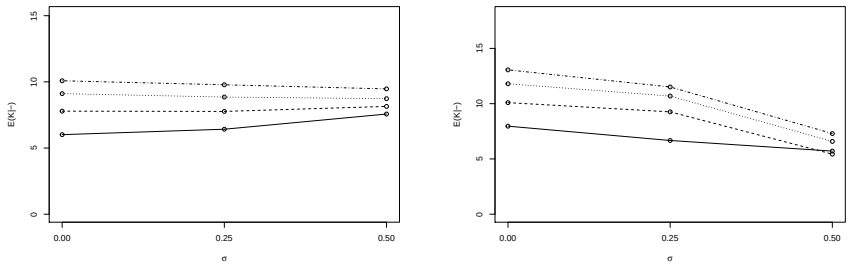
Per fornire una rappresentazione grafica dei fenomeni appena descritti, in Figura 2.1, 2.2, 2.3, 2.4 sono rappresentate le medie a posteriori del numero di gruppi (in ordinata) al variare di σ (in ascissa) per le diverse specificazioni di θ. Sono riportati solo i grafici per gli scenari corrispondenti a una numerosità $n = 70$, poiché il comportamento è analogo per le altre numerosità campionarie considerate. Le diverse linee indicano le diverse specificazioni a priori del numero atteso di gruppi $\mathbb{E}[K_n]$: la linea continua corrisponde a 5, quella tratteggiata a 10, quella punteggiata a 15 e la mista a 20.

Più che il numero di componenti, l'interesse è di valutare la stima della distribuzione di probabilità. Per fare ciò, di seguito (Figg. 2.5, 2.6, 2.7, 2.8) sono riportate le medie a posteriori della distribuzione di probabilità ottenute



(a) Mistura di Poisson (b) Mistura di RG

Figura 2.3: Pólya urn Gibbs sampler: media a posteriori del numero di componenti. Dati simulati da una mistura di 3 *Rounded Gaussian*.

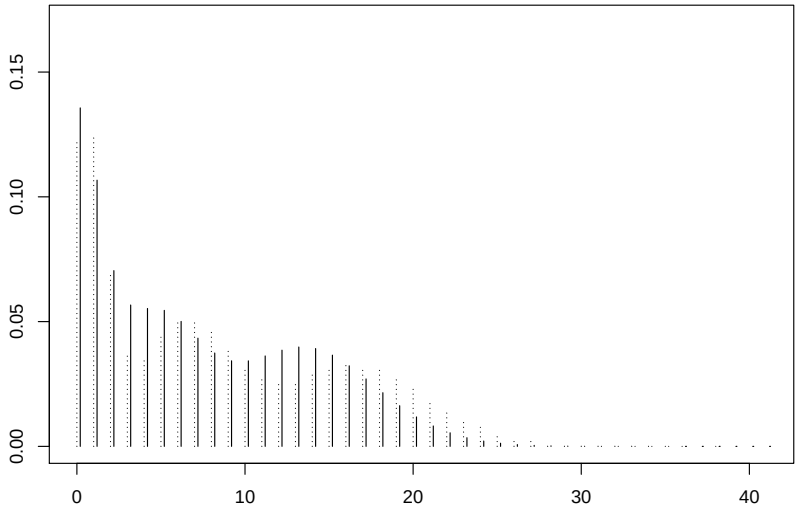


(a) Mistura di Poisson (b) Mistura di RG

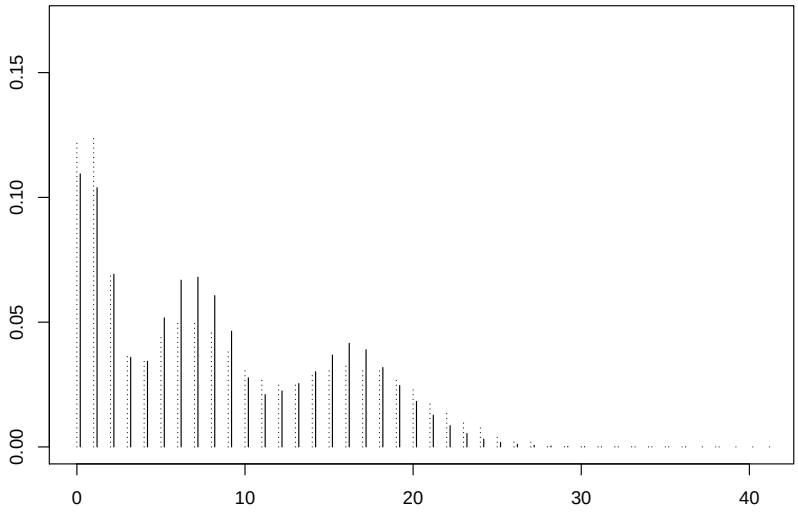
Figura 2.4: Pólya urn Gibbs sampler: media a posteriori del numero di componenti. Dati simulati da una mistura di 6 *Rounded Gaussian*.

con i due diversi *kernel* per gli scenari con numerosità $n = 70$. Sono riportate solo le distribuzioni stimate per $\sigma = 0.5$ e un numero di *cluster* atteso a priori pari a 15, poiché i risultati per le altre combinazioni di parametri non presentano differenze sostanziali; con la linea punteggiata è riportata la vera distribuzione da cui sono simulati i dati. Le stime della distribuzione sono ottenute come descritto da MacEachern e Müller (1998).

Dai grafici risultano evidenti i problemi del *kernel* Poisson descritti nella Sezione 1.7. In particolare, negli scenari simulati da una mistura di RG, si nota l'impossibilità della Poisson di rappresentare componenti che presentano sottodispersione. Al contrario, la mistura di RG dà buoni risultati anche nel caso sia applicata a uno scenario simulato dalla Poisson.

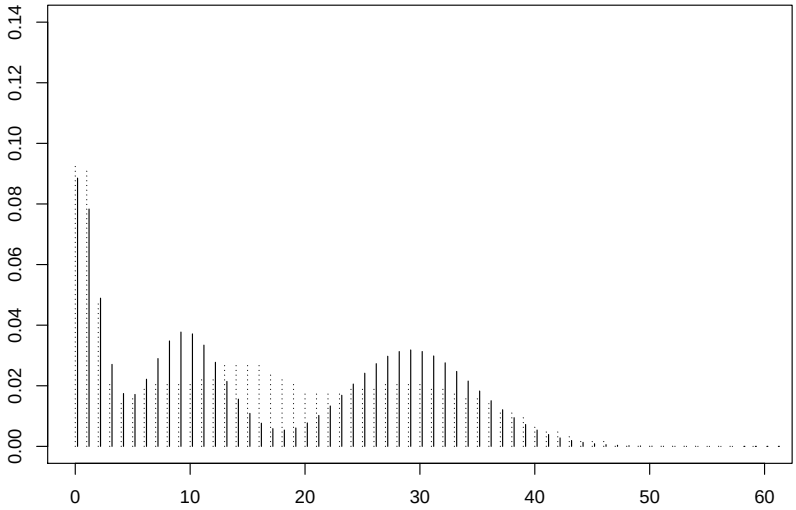


(a) Mistura di Poisson

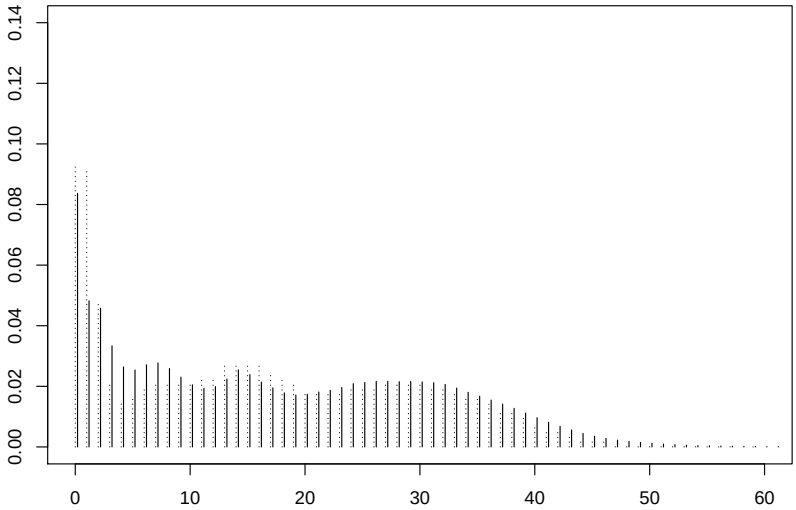


(b) Mistura di RG

Figura 2.5: Distribuzione predittiva a posteriori. Dati simulati da una mistura di 3 Poisson.

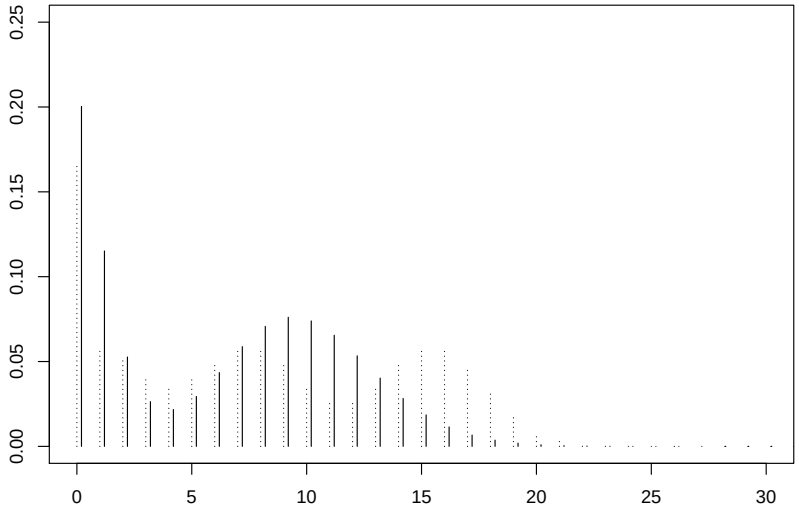


(a) Mistura di Poisson

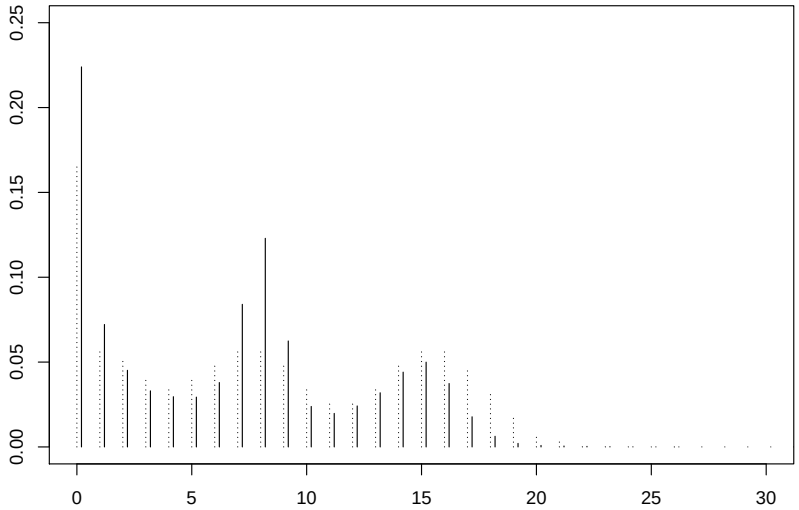


(b) Mistura di RG

Figura 2.6: Distribuzione predittiva a posteriori. Dati simulati da una mistura di 6 Poisson.

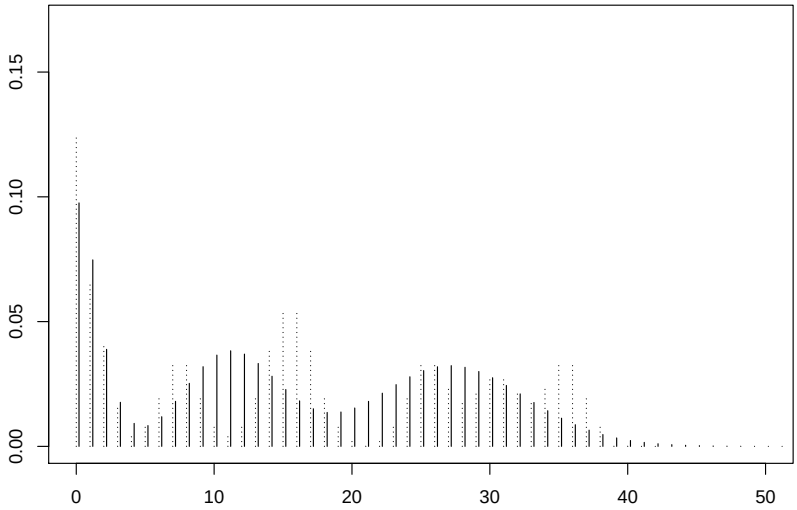


(a) Mistura di Poisson

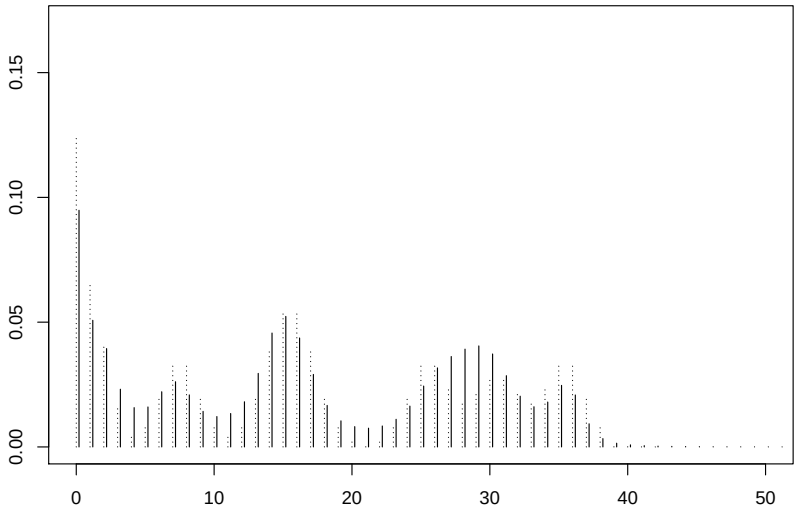


(b) Mistura di RG

Figura 2.7: Distribuzione predittiva a posteriori. Dati simulati da una mistura di 3 *Rounded Gaussian*.



(a) Mistura di Poisson



(b) Mistura di RG

Figura 2.8: Distribuzione predittiva a posteriori. Dati simulati da una mistura di 6 *Rounded Gaussian*.

Capitolo 3

Applicazione: data set Norberg

3.1 I dati

Il data set analizzato consiste in un portfolio di assicurazioni sulla vita collettive: queste sono particolari polizze che vengono stipulate da un unico contraente per assicurare un gruppo di persone; il caso più tipico è quello di un datore di lavoro che assicura i dipendenti. Diversamente dalle assicurazioni sulla vita individuali, in questo contesto il contraente, ovvero l'azienda o il datore di lavoro, può verificare se nel lungo periodo il premio pagato è comparabile ai risarcimenti ottenuti. Per questo motivo è di particolare interesse stimare nel modo più preciso possibile l'ammontare totale che l'assicurazione dovrà pagare ad un particolare gruppo. Questo dipenderà, ovviamente, dal numero totale di richieste di risarcimento, e da qui nasce il problema in esame.

Il data set Norberg, disponibile nel pacchetto R `REBayes`, consiste di 72 gruppi di lavoratori, per ciascuno dei quali sono registrati il numero di morti (quindi di richieste di risarcimento) e la relativa esposizione al rischio. Questa è calcolata come la somma dell'esposizione di tutti gli individui appartenenti al gruppo, espressa in anni, e può essere interpretata come funzione del numero di eventi che ci si aspetta a priori. Il data set nasce dal raggruppamento di 1125 gruppi di assicurati considerati omogenei rispetto a caratteristiche di rischio osservabili, come descritto da Norberg (1989).

Il numero di richieste di risarcimento è compreso tra 0 e 57, mentre l'esposizione varia in un range di valori compreso tra 4.45 e 26762.37: il numero medio di morti per unità di esposizione è pari a 0.003, con varianza 1.98×10^{-5}. Per rendere le variabili più confrontabili, Haastrup (2000) normalizza l'esposizione per un fattore pari a 344, in modo che il numero medio di morti, normalizzato per l'esposizione, sia pari a 1 (più precisamente 1.11),

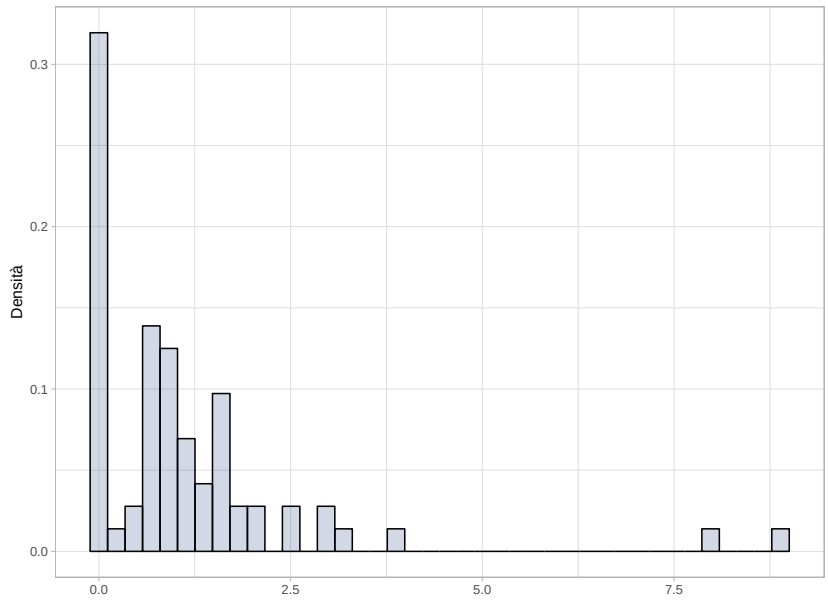


Figura 3.1: Istogramma del numero di decessi per unità di esposizione.

mentre la varianza è pari a 2.35. In Figura 3.1 è riportato l'istogramma del numero di decessi per unità di esposizione: una caratteristica evidente è la forte presenza di gruppi per cui il numero di eventi è pari a zero. La maggior parte dei restanti gruppi si concentra intorno a valori piccoli ($1-2$ morti per unità di esposizione), tuttavia, ci sono pochi gruppi isolati per cui il numero di decessi è piuttosto elevato.

Vista la forma della distribuzione, definire un modello che riesca a descrivere adeguatamente i dati risulta problematico. Inoltre, in questo contesto, per questioni economiche, in genere non vengono rilevate molte caratteristiche e fattori di rischio che sarebbero importanti per una stima adeguata del numero di decessi, comportando la presenza di un'eterogeneità non osservata non trascurabile (Norberg, 1989).

3.2 Modelli proposti

Per modellare questa eterogeneità, Haastrup (2000) propone un modello gerarchico bayesiano parametrico in cui ogni gruppo è caratterizzato da un diverso parametro λ_i, estratto indipendentemente da una distribuzione gamma, che rappresenta il fattore di rischio latente. Condizionatamente al para-

metro, ogni gruppo è assunto indipendente e il numero di richieste di risarcimento è modellato con una distribuzione Poisson: per tenere conto della diversa esposizione al rischio è intuitivo utilizzare una $Poisson(\lambda_i E_i)$, dove E_i è l'esposizione dell'i-mo gruppo. Il modello può, quindi, essere espresso come:

$$\begin{aligned} Y_i|\lambda_i &\sim Poisson(\lambda_i E_i) \\ \lambda_i &\overset{iid}{\sim} Gamma(\alpha, \beta) \end{aligned} \tag{3.1}$$

dove su α e β sono poste delle distribuzioni a priori $\pi(\alpha)$ e $\pi(\beta)$.

Dal momento che i parametri λ_i non sono osservabili, imporre una specifica distribuzione per descrivere l'eterogeneità tra i gruppi può risultare troppo restrittivo. Per questo motivo, Haastrup (2000) propone anche una formulazione nonparametrica del modello basata sul processo di Dirichlet, tuttavia, non evidenzia differenze sostanziali rispetto al modello 3.1: in particolare, non identifica un numero limitato di categorie di rischio e mantiene l'assunzione che si tratti di un'unica popolazione eterogenea. Un modello mistura analogo è in seguito ripreso da Brown e Buckley (2015), che si concentrano, invece, proprio sull'identificazione del numero di gruppi presenti nel campione. Contrariamente da Haastrup (2000), evidenziano come l'ipotesi di uguale eterogeneità tra i gruppi non sia verificata, poiché la probabilità a posteriori si concentra su modelli con sole due o tre componenti.

Di seguito si considereranno due diverse formulazioni di modelli mistura e se ne confronteranno i risultati: la prima è una mistura nonparametrica di *kernel* Poisson con distribuzione misturante DP, simile a quella proposta da Haastrup (2000) e Brown e Buckley (2015), ovvero:

$$\begin{aligned} Y_i|\lambda_i &\sim Poisson(\lambda_i E_i) \\ \lambda_i|P &\overset{iid}{\sim} P \\ P &\sim DP(\alpha, P_0) \end{aligned} \tag{3.2}$$

dove a P_0 corrisponde una distribuzione $Gamma(a, b)$. Il parametro α del DP è stato scelto in modo da avere, a priori, un numero di gruppi atteso pari a 10 (corrisponde ad un valore α circa pari a 3) per cercare di ottenere risultati il più possibile simili a Brown e Buckley (2015). La loro formulazione del modello si basa, infatti, su una distribuzione a priori $Dir(\delta_1, \ldots, \delta_K)$ con $\delta_k = 1$ per ogni $k = 1, \ldots, K$. Tuttavia, poiché la dimensione K varia, non è possibile determinare un unico parametro α che definisca il DP corrispondente.

La distribuzione usata per l'aggiornamento di λ nel Gibbs sampler (corrisponde al passo (b) dell'algoritmo esposto nella Sezione 1.6.2) è pari a:

$$\lambda_h \sim Gamma(a + n_h \bar{y}_h, b + n_h \bar{E}_h)$$

dove n_h è il numero di osservazioni correntemente allocate al *cluster* h, $\bar{y}_h = n_h^{-1} \sum_{i:c_i=h} y_i$ e $\bar{E}_h = n_h^{-1} \sum_{i:c_i=h} E_i$.

La seconda formulazione si basa su un *kernel* RG e un processo di Pitman-Yor. Non esistendo per il *kernel* RG un parametro analogo al parametro di tasso della Poisson, l'esposizione è stata introdotta come fattore moltiplicativo sulla media delle variabili latenti: in questo modo il valore atteso del numero di eventi è proporzionale alla durata dell'esposizione al rischio. Sulla varianza, invece, non si sono imposte restrizioni ed è quindi lasciata indipendente dall'esposizione. Il modello può essere scritto come:

$$\begin{aligned} Y_i|\mu_i, \tau_i &\sim RG(\mu_i E_i, \tau_i^{-1}) \\ (\mu_i, \tau_i)|P &\overset{iid}{\sim} P \\ P &\sim PY(\sigma, \theta, P_0) \end{aligned} \tag{3.3}$$

con $P_0 = N(\mu; \mu_0, \kappa\tau^{-1})Gamma(\tau; \alpha, \beta)$. Per i risultati visti nel capitolo precedente, il parametro σ è stato posto pari a 0.50, mentre θ è stato scelto in modo da avere, a priori, un numero di gruppi atteso pari a 10, per uniformità con il modello precedente.

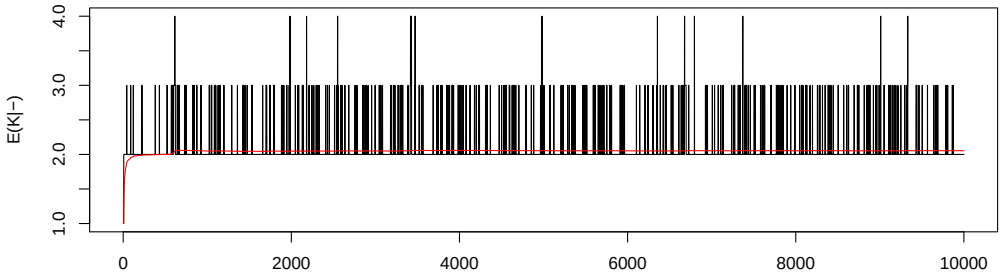
Anche in questo caso l'inserimento dell'esposizione comporta una modifica della distribuzione per l'aggiornamento dei parametri (μ, τ) all'interno del Gibbs sampler, che diventa:

$$(\mu_h, \tau_h^{-1}) \sim N(\hat{\mu}_h, \hat{\kappa}_h \hat{\tau}_h^{-1})Gamma(\hat{\alpha}_h, \hat{\beta}_h)$$

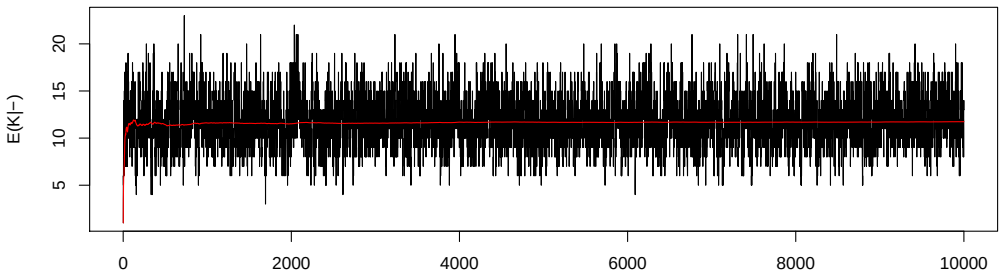
con $\hat{\kappa}_h = \kappa^{-1} + \sum_{i:c_i=h} E_i^2$; $\hat{\mu}_h = \hat{\kappa}_h(\kappa^{-1}\mu_0 + \sum_{i:c_i=h} E_i y_i)$; $\hat{\alpha}_h = \alpha + n_h/2$; $\hat{\beta}_h = \beta - \frac{1}{2}[(\hat{\kappa}_h - 1)\kappa^{-1}\mu_0^2 - \sum_{i:c_i=h} y_i^2 + \hat{\kappa}_h(2\mu_0 \sum_{i:c_i=h} E_i y_i + (\sum_{i:c_i=h} E_i y_i)^2)]$.

Riguardo all'esposizione, si è considerata sia nella scala originale, come fatto da Brown e Buckley (2015), sia divisa per 344 come proposto da Haastrup (2000). Poiché ciò che cambia tra le due diverse specificazioni è solo l'unità di misura con cui è espressa tale variabile, il risultato dell'inferenza con l'una o l'altra specificazione dovrebbe essere lo stesso.

Per tutti i modelli si sono simulate, tramite Pólya urn Gibbs sampler, catene di lunghezza 10000, con un burn-in di 3000. Per verificarne la convergenza, oltre all'analisi grafica (si veda l'Appendice A per i traceplot delle catene), si è utilizzata la diagnostica di Geweke (disponibile nel pacchetto R "coda"). Questa diagnostica è basata su un test di uguaglianza delle medie della prima e dell'ultima parte della catena: anziché riportare direttamente i risultati dei test, i valori osservati sono stati rappresentati graficamente al fine di renderne più immediata l'interpretazione. Poiché la statistica test segue una distribuzione $N(0, 1)$, nei grafici sono rappresentate anche due linee



(a) Esposizione nella scala originale



(b) Esposizione normalizzata

Figura 3.2: Mistura di Poisson: traceplot del numero di componenti.

tratteggiate corrispondenti ai quantili 0.025 e 0.975 della normale standard, per dare un'indicazione dei valori attesi sotto ipotesi di convergenza.

3.3 Confronto tra modelli

3.3.1 Mistura di *kernel* Poisson

Il primo modello analizzato è la mistura di *kernel* Poisson con l'esposizione espressa nella scala originale. Per verificare la convergenza delle catene ottenute, in Figura A.1 e A.2 sono riportati i traceplot delle probabilità a posteriori $p(j|y_1, \ldots, y_n)$ per $j = 0, 1, 2, 3, 4, 5$, dove l'esposizione è stata fis-

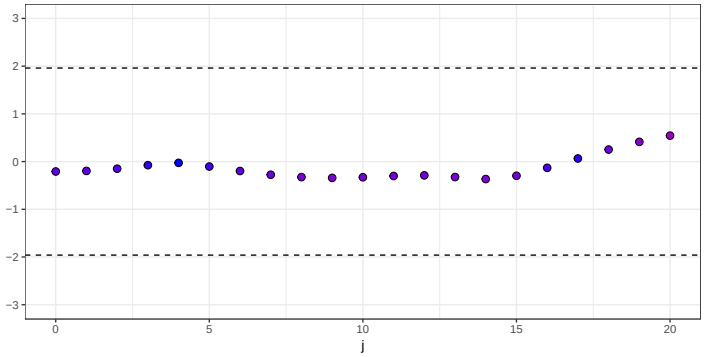
sata pari a 1000; mentre in Figura 3.3(a) sono riportate le corrispondenti diagnostiche di Geweke per $j = 0, \ldots, 20$.

Poiché il metodo usato per stimare questo modello (Pólya urn Gibbs sampler) è molto diverso da quello usato da Brown e Buckley (2015), i risultati ottenuti non sono del tutto uguali. In particolare, attraverso il RJMCMC da loro proposto, il numero di componenti stimato è pari a 2 o 3 con probabilità 0.88, mentre la restante probabilità è assegnata a modelli con 4 o 5 componenti. Le catene ottenute tramite Pólya urn Gibbs sampler, invece, oscillano solo tra modelli con 2 e 3 componenti, e quasi mai identificano un numero di componenti maggiore, come si può notare dal traceplot riportato in Figura 3.2(a).

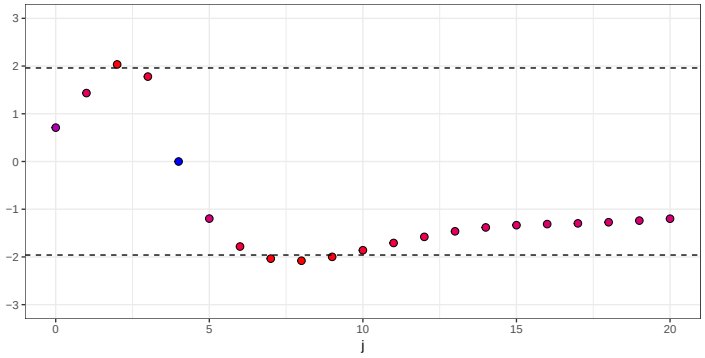
Si è poi stimato lo stesso modello, ma con l'esposizione divisa per 344: il traceplot del numero di componenti stimato è riportato in figura 3.2(b); mentre i traceplot delle probabilità stimate e le diagnostiche di Geweke sono riportate, rispettivamente, in Figura A.3 e in Figura 3.3(b). Per ottenere risultati confrontabili, per la stima di $p(j|y)$ l'esposizione si è fissata pari a 1000/344.

Dai traceplot del numero di componenti (Fig. 3.2) si nota un fenomeno piuttosto spiacevole: a seconda della scala in cui è espressa l'esposizione, le catene convergono a valori molto diversi. In particolare, la media a posteriori $\mathbb{E}[K|y]$ stimata con l'esposizione non normalizzata è pari a 2.05, mentre è pari a 11.80 nel caso venga divisa per 344.

La differenza tra i risultati ottenuti dalle due diverse specificazioni è evidenziata anche dalle distribuzioni predittive a posteriori che ne derivano, come mostrato in Figura 3.4 (per i casi con l'esposizione normalizzata, il valore usato per ricavare le stime è pari a quello indicato diviso per 344). Mentre per esposizioni basse non si nota una differenza sostanziale, per valori elevati le distribuzioni stimate sono decisamente diverse: in particolare, la distribuzione stimata senza normalizzazione è fortemente bimodale, mentre quella ottenuta normalizzando è solo asimmetrica.

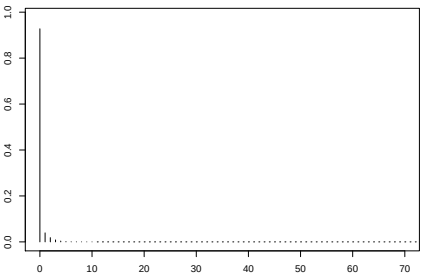


(a) Esposizione originale

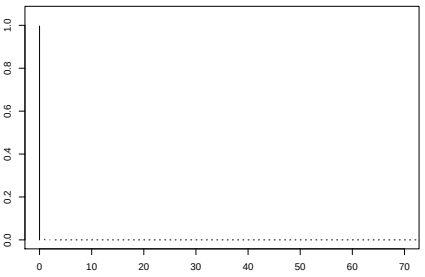


(b) Esposizione normalizzata

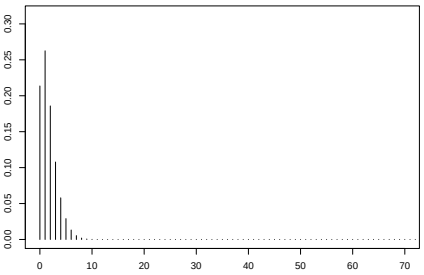
Figura 3.3: Diagnostiche di Geweke per $p(j|y)$ con mistura Poisson: valore della statistica z (in ordinata) per $j = 0, \ldots, 20$ (in ascissa).



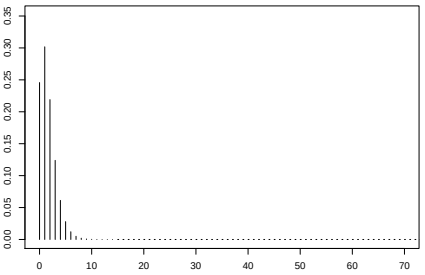
(a) E=1



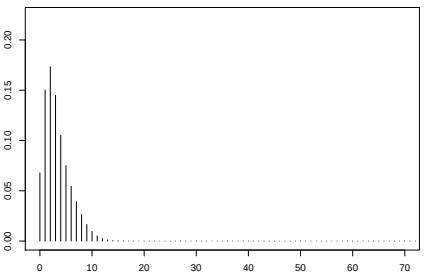
(b) E=1



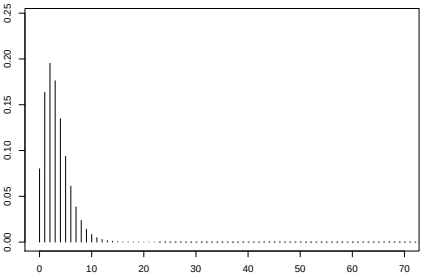
(c) E=500



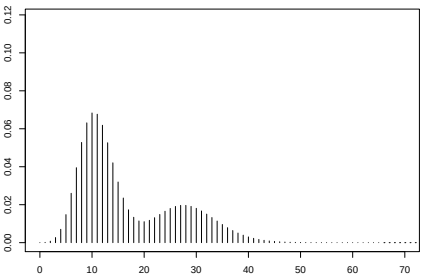
(d) E=500



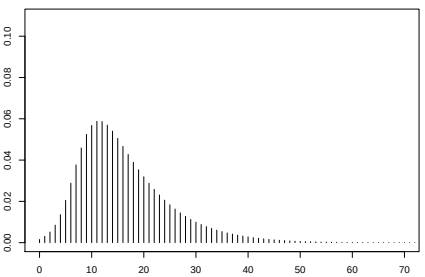
(e) E=1000



(f) E=1000

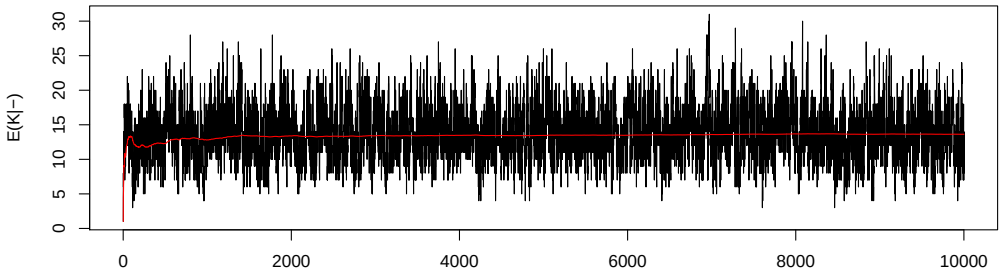


(g) E=5000

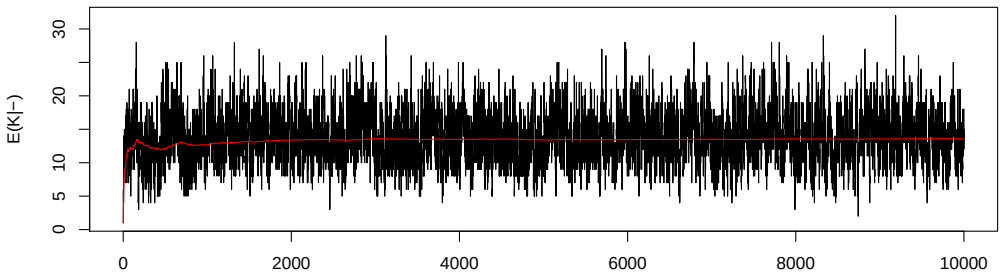


(h) E=5000

Figura 3.4: Mistura di Poisson: distribuzione predittiva a posteriori. A sinistra risultati con l'esposizione originale, a destra normalizzata.



(a) Esposizione nella scala originale



(b) Esposizione normalizzata

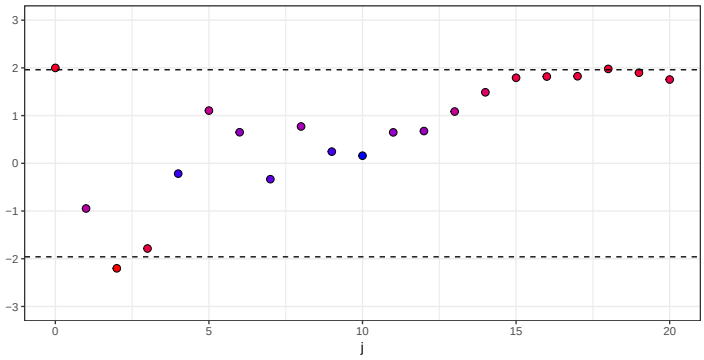
Figura 3.5: Mistura di RG: traceplot del numero di componenti.

3.3.2 Mistura di *kernel* RG

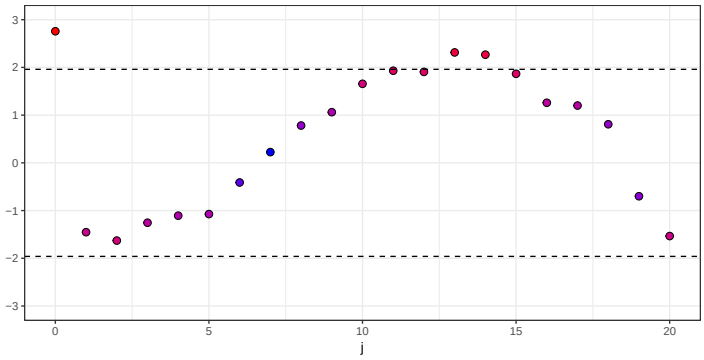
Si analizzeranno ora i risultati della stima del modello basato sul *kernel Rounded Gaussian*. Analogamente a quanto fatto per la Poisson, sono riportati i traceplot del numero di componenti (Fig. 3.5) e delle probabilità $p(j|y)$ per $j = 0, \ldots, 5$ (Figg. A.4, A.5), insieme alle corrispondenti diagnostiche di Geweke (Fig. 3.6).

Un primo risultato evidente è che la stima del numero di componenti ora è decisamente più coerente rispetto alla diversa specificazione dell'esposizione: nel caso si consideri la scala originale la media è pari a 13.72, mentre è pari a 13.56 con l'esposizione normalizzata.

Le distribuzioni predittive a posteriori sono riportate in Figura 3.7: le distribuzioni stimate con le due diverse scale per l'esposizione sono pratica-



(a) Esposizione originale



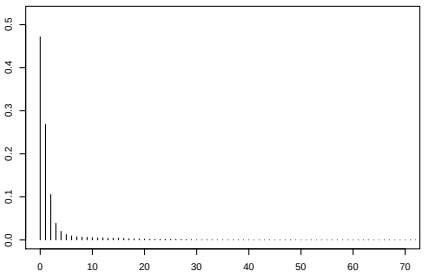
(b) Esposizione normalizzata

Figura 3.6: Diagnostiche di Geweke per $p(j|y)$ con mistura RG: valore della statistica z (in ordinata) per $j = 0, \ldots, 20$ (in ascissa).

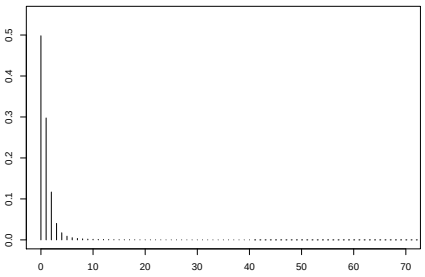
mente identiche. Una caratteristica presente in tutte le distribuzioni, anche per esposizioni grandi, è la grande massa di probabilità stimata in zero. Questo è comunque ragionevole se si considerano i dati a disposizione, poiché ci sono svariati gruppi con esposizioni non trascurabili e nessun decesso.

3.3.3 Conclusioni

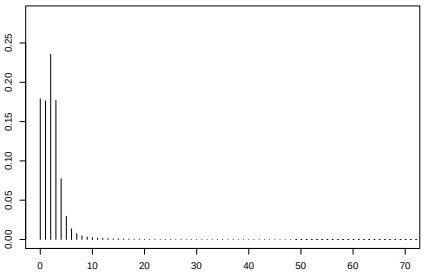
Visti i risultati ottenuti, appare evidente come la mistura di *kernel* Poisson risulti problematica: in particolare, il fatto di ottenere risultati così diversi a seconda della scala in cui è espressa l'esposizione fa sorgere dei dubbi riguardo a tutta l'inferenza che ne deriva. Al contrario, la mistura di *kernel* RG fornisce risultati ragionevoli e stabili rispetto alla specificazione usata. Sempre riguardo all'esposizione, per indagare meglio il fenomeno osservato per il modello Poisson, si è provato a stimare anche una mistura RG con



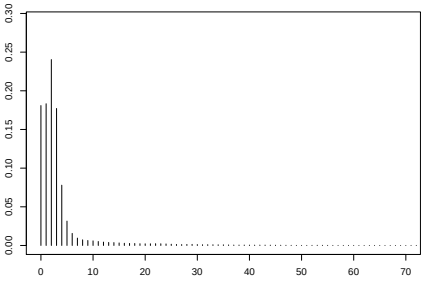
(a) E=1



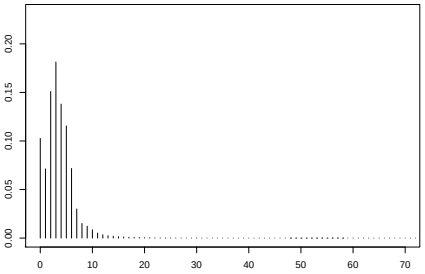
(b) E=1



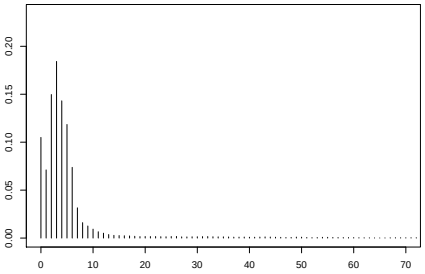
(c) E=500



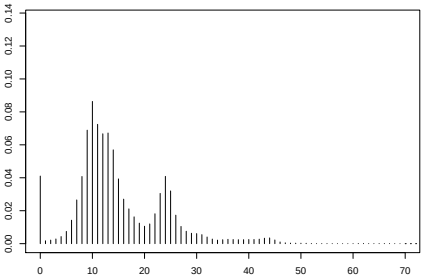
(d) E=500



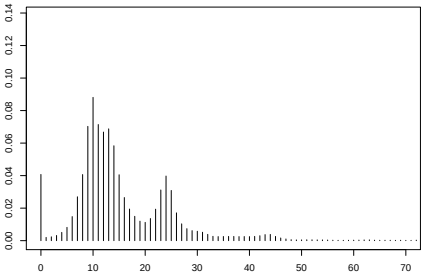
(e) E=1000



(f) E=1000



(g) E=5000



(h) E=5000

Figura 3.7: Mistura di RG: distribuzione predittiva a posteriori. A sinistra risultati con l'esposizione originale, a destra normalizzata.

l'esposizione introdotta come fattore moltiplicativo sulla varianza, in modo da avere i primi due momenti uguali alla distribuzione Poisson. In questo caso si osservava lo stesso comportamento: sia il numero di componenti, sia le distribuzioni predittive ottenute risultavano diverse a seconda della specificazione usata per l'esposizione. Questo fa pensare che, nel caso del modello Poisson, il problema nasca proprio dalla struttura troppo rigida del *kernel* Poisson, che costringe la varianza ad essere uguale alla media.

Conclusioni

In questa tesi si è proposto un modello bayesiano nonparametrico per la stima della distribuzione del numero di richieste di risarcimento nell'ambito delle assicurazioni sulla vita collettive. In particolare, ci si è concentrati sulla specificazione di un modello mistura che fosse adatto ai particolari dati in esame, costituiti da conteggi. A questo scopo si sono individuate due possibili formulazioni, basate su diversi *kernel*: quello Poisson e *Rounded Gaussian*. Oltre alla scelta del *kernel*, si sono valutate anche due alternative per il processo stocastico usato come distribuzione misturante, ovvero il processo di Dirichlet e quello di Pitman-Yor.

Per quanto riguarda la scelta del processo, attraverso la simulazione si è constatata l'impossibilità di utilizzare algoritmi condizionali per stimare misture nonparametriche basate sul processo di Pitman-Yor, a causa del numero eccessivamente elevato di componenti necessarie per rappresentarlo. Nonostante questa difficoltà, sia le motivazioni teoriche sia le simulazioni portano a preferire questo processo rispetto al DP, per la robustezza delle conclusioni rispetto alla specificazione a priori.

Riguardo alla scelta *kernel*, invece, si è visto come le misture basate sul *kernel* Poisson ereditino molte problematiche legate a questa distribuzione e, addirittura, ne introducano altre. Nasce proprio dalla struttura troppo rigida del *kernel*, infatti, il fenomeno che si è osservato all'aumentare del parametro σ del processo di Pitman-Yor, per cui le misture tendevano ad introdurre un numero sempre maggiore di componenti. Al contrario, le misture basate sul *kernel Rounded Gaussian*, oltre a godere di buone proprietà teoriche, hanno fornito risultati positivi in tutti gli scenari considerati in simulazione.

Sulla base di queste considerazioni, si è scelto di modellare i dati in esame per mezzo di una mistura nonparametrica di *kernel Rounded Gaussian* basata sul processo di Pitman-Yor. Anche in questo caso il modello così formulato ha fornito risultati decisamente preferibili rispetto alla mistura basata sul *kernel* Poisson. In particolare, si è visto come la struttura troppo rigida di questa distribuzione portasse a risultati contraddittori a seconda della specificazione usata per esprimere la durata dell'esposizione al rischio

dei soggetti. Al contrario, la mistura di *kernel Rounded Gaussian*, oltre a fornire risultati coerenti in tutte le specificazioni considerate, si è rivelata anche particolarmente adatta ai dati analizzati per la capacità di identificare distribuzioni che presentano una grande massa di probabilità in zero.

Appendice A

Materiale aggiuntivo

A.1 Risultati delle simulazioni

Di seguito sono riportate le tabelle con i risultati degli algoritmi descritti nel Capitolo 2.

Slice Gibbs sampler

Scenario 1: mistura di kernel Poisson

n	k_0	$\mathbb{E}[K_n]$	$\sigma = 0$	max_c	$\sigma = 0.25$	max_c	$\sigma = 0.50$	max_c
50	3	5	8.03	30	7.52	298	9.23	500
		10	10.06	48	10.00	411	10.17	498
		15	12.53	113	14.85	496	12.60	500
		20	14.79	183	14.36	488	15.21	500
	6	5	6.07	36	7.01	244	8.43	500
		10	9.05	51	8.45	467	9.20	500
		15	11.02	99	10.94	500	10.29	500
		20	12.19	183	12.31	489	12.88	500
70	3	5	6.56	27	6.96	185	9.83	500
		10	9.82	55	9.17	413	10.50	498
		15	12.32	83	11.75	493	13.63	500
		20	14.19	130	13.71	492	13.13	498
	6	5	7.04	27	6.90	121	8.18	500
		10	11.51	59	8.35	450	10.28	500
		15	11.72	82	9.83	493	9.92	499
		20	12.24	130	11.57	500	10.93	500
100	3	5	6.93	23	9.30	443	12.46	499
		10	9.68	49	10.36	377	12.44	500
		15	13.26	74	13.47	498	15.38	500
		20	14.73	116	16.79	499	14.41	499
	6	5	7.32	25	10.47	320	10.45	500
		10	10.29	41	10.32	472	11.06	500
		15	11.61	78	11.54	496	16.54	500
		20	13.54	114	15.33	461	12.92	500

Tabella A.1: Risultati dello slice Gibbs sampler con distribuzione Poisson

Scenario 2: mistura di kernel *Rounded Gaussian*

n	k_0	$\mathbb{E}[K_n]$	$\sigma = 0$	max_c	$\sigma = 0.25$	max_c	$\sigma = 0.50$	max_c
50	3	5	6.40	32	7.05	183	8.36	500
		10	9.47	64	13.69	484	10.43	500
		15	12.04	101	12.82	484	11.81	500
		20	14.15	164	13.85	487	13.23	500
	6	5	6.49	28	7.01	206	8.88	500
		10	8.94	53	12.31	493	14.87	500
		15	10.42	102	10.41	490	10.12	500
		20	12.05	161	12.64	499	11.19	500
70	3	5	6.78	27	6.80	295	9.39	500
		10	9.56	57	9.78	368	10.56	500
		15	12.97	86	12.19	489	12.85	500
		20	15.03	144	14.45	497	14.19	500
	6	5	6.15	23	6.31	430	7.78	499
		10	8.33	60	7.77	381	8.16	500
		15	9.35	81	11.81	486	10.61	500
		20	11.55	137	10.70	458	9.64	500
100	3	5	6.72	27	7.91	315	10.55	499
		10	11.34	50	10.58	316	12.75	496
		15	13.08	64	13.19	496	13.08	500
		20	15.24	105	18.31	499	16.56	500
	6	5	6.73	27	9.43	259	10.86	500
		10	10.95	48	11.21	478	12.01	499
		15	11.93	76	11.99	462	12.16	500
		20	15.04	120	14.40	491	13.93	500

Tabella A.2: Risultati dello slice Gibbs sampler con distribuzione Poisson

Scenario 1: mistura di kernel Poisson

n	k_0	$\mathbb{E}[K_n]$	$\sigma = 0$	max_c	$\sigma = 0.25$	max_c	$\sigma = 0.50$	max_c
50	3	5	10.96	32	14.55	398	19.92	500
		10	15.04	62	16.98	473	19.88	500
		15	18.83	107	19.62	483	21.17	500
		20	22.57	164	23.10	499	23.12	500
	6	5	12.37	31	16.51	486	20.87	500
		10	16.03	59	18.19	460	21.93	500
		15	19.26	105	21.11	482	22.72	500
		20	22.46	168	23.06	500	23.94	500
70	3	5	10.33	35	15.20	413	21.19	500
		10	14.89	70	17.30	475	22.47	500
		15	18.89	104	20.94	499	23.12	500
		20	23.21	135	24.54	499	25.63	500
	6	5	14.84	33	21.79	493	29.53	500
		10	18.42	63	23.56	497	29.85	500
		15	22.56	94	25.82	500	30.28	500
		20	26.18	137	28.49	499	31.40	500
100	3	5	8.81	32	18.32	415	32.39	500
		10	17.02	52	22.74	498	31.33	500
		15	22.06	86	24.52	499	32.60	500
		20	24.97	116	28.35	500	33.02	500
	6	5	11.79	30	24.50	483	34.13	500
		10	19.18	56	24.63	499	36.13	500
		15	23.31	121	28.19	485	36.50	500
		20	27.68	122	32.22	499	37.15	500

Tabella A.3: Risultati dello slice Gibbs sampler con distribuzione RG

Scenario 2: mistura di kernel *Rounded Gaussian*

n	k_0	$\mathbb{E}[K_n]$	$\sigma = 0$	max_c	$\sigma = 0.25$	max_c	$\sigma = 0.50$	max_c
50	3	5	9.18	32	11.99	436	16.34	500
		10	13.12	78	14.70	497	16.54	500
		15	16.89	107	17.87	475	18.01	500
		20	20.78	192	20.94	500	20.13	500
	6	5	11.59	39	15.17	437	19.59	500
		10	15.08	66	16.84	488	20.26	500
		15	18.49	125	19.47	500	21.73	500
		20	21.92	160	22.19	495	23.10	500
70	3	5	9.23	35	11.96	497	17.29	500
		10	13.30	52	15.06	446	17.67	500
		15	17.69	100	18.40	494	20.08	500
		20	21.78	143	21.78	499	21.92	500
	6	5	12.83	31	18.53	471	25.26	500
		10	17.08	59	20.80	481	25.69	500
		15	20.77	96	22.75	472	26.07	500
		20	23.97	133	26.04	500	27.49	500
100	3	5	9.69	26	13.86	423	20.12	500
		10	13.97	46	15.52	499	21.00	500
		15	17.21	79	18.85	500	22.63	500
		20	21.30	107	22.54	494	23.39	500
	6	5	12.18	33	16.89	457	22.75	500
		10	15.14	49	19.76	497	24.41	500
		15	19.22	85	21.91	472	26.02	500
		20	22.57	100	24.38	491	25.29	500

Tabella A.4: Risultati dello slice Gibbs sampler con distribuzione RG

Pólya urn Gibbs sampler

			Scenario 1: Poisson			Scenario 2: RG		
n	k_0	$\mathbb{E}[K_n]$	$\sigma = 0$	$\sigma = 0.25$	$\sigma = 0.50$	$\sigma = 0$	$\sigma = 0.25$	$\sigma = 0.50$
50	3	5	6.74	7.54	8.87	6.71	7.77	9.58
		10	9.69	9.89	10.37	9.40	9.25	10.71
		15	12.19	11.90	11.77	11.28	11.10	11.33
		20	14.13	13.95	13.47	13.12	12.56	12.44
	6	5	6.33	6.76	8.09	6.29	6.71	7.80
		10	8.70	8.64	9.10	8.49	8.31	8.72
		15	10.45	10.24	10.30	10.15	9.80	9.69
		20	11.81	11.61	11.33	11.45	11.11	10.85
70	3	5	6.64	7.33	9.72	6.53	7.23	9.21
		10	9.43	9.51	10.70	9.61	9.67	10.29
		15	11.78	11.35	11.46	12.31	11.90	12.03
		20	13.75	13.33	12.88	14.57	14.05	13.31
	6	5	6.23	6.62	7.66	6.01	6.42	7.57
		10	8.27	8.11	8.59	7.79	7.77	8.15
		15	9.96	9.59	9.39	9.11	8.85	8.73
		20	11.35	10.76	10.26	10.08	9.78	9.47
100	3	5	6.77	7.90	10.86	6.63	7.75	10.35
		10	9.79	9.92	12.22	9.84	10.06	11.87
		15	12.56	12.44	13.26	12.65	12.26	13.06
		20	14.83	14.43	14.42	14.88	14.31	14.12
	6	5	6.71	7.77	9.58	6.96	7.96	10.31
		10	9.40	9.25	10.71	9.34	9.86	11.00
		15	11.28	11.10	11.33	11.48	11.54	11.94
		20	13.12	12.56	12.44	13.34	12.87	13.03

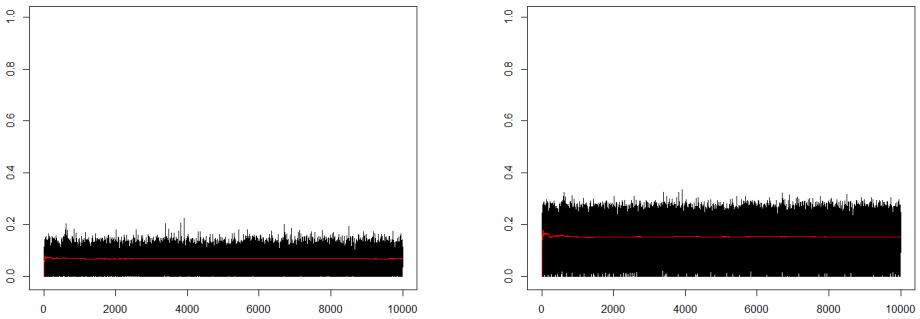
Tabella A.5: Risultati del Pólya urn Gibbs sampler con distribuzione Poisson

			Scenario 1: Poisson			Scenario 2: RG		
n	k_0	$\mathbb{E}[K_n]$	$\sigma = 0$	$\sigma = 0.25$	$\sigma = 0.50$	$\sigma = 0$	$\sigma = 0.25$	$\sigma = 0.50$
50	3	5	7.41	7.70	4.57	7.18	7.42	7.43
		10	9.32	8.48	6.49	9.40	8.81	8.03
		15	10.77	9.70	8.55	11.12	9.99	8.75
		20	12.03	10.85	8.92	12.28	11.06	9.61
	6	5	6.98	5.69	5.08	7.21	7.40	5.89
		10	8.38	7.41	5.22	8.63	8.16	7.03
		15	9.02	8.00	5.86	9.75	8.88	7.29
		20	10.01	9.11	6.71	10.77	9.29	7.86
70	3	5	7.41	8.04	8.56	7.58	8.46	8.62
		10	10.26	10.05	8.43	10.31	9.75	9.31
		15	12.49	11.04	8.77	12.57	11.49	10.10
		20	14.46	12.83	10.57	14.53	13.11	10.94
	6	5	5.14	4.49	3.19	7.97	6.67	5.71
		10	8.97	6.81	4.63	10.09	9.26	5.43
		15	10.95	8.09	5.21	11.80	10.69	6.59
		20	11.84	9.97	5.79	13.06	11.51	7.30
100	3	5	6.04	6.06	5.59	7.52	8.32	7.83
		10	9.54	7.99	6.04	10.37	9.31	8.84
		15	11.74	9.66	7.29	12.51	10.76	9.16
		20	14.28	11.13	7.83	14.40	12.46	9.60
	6	5	6.66	5.92	5.04	7.84	8.42	8.60
		10	9.63	8.08	5.14	10.48	9.95	8.79
		15	12.07	9.11	6.63	13.03	11.39	9.21
		20	13.65	10.96	7.01	14.80	12.67	9.88

Tabella A.6: Risultati del Pólya urn Gibbs sampler con distribuzione *Rounded Gaussian*

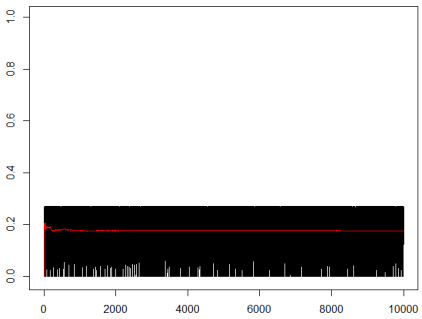
A.2 Applicazione: catene simulate

Di seguito sono riportati i traceplot delle catene di $p(j|y)$, per $j = 0, 1, 2, 3, 4, 5$, per i modelli descritti nel Capitolo 3. In rosso è mostrata la stima della media ottenuta al crescere delle iterazioni.

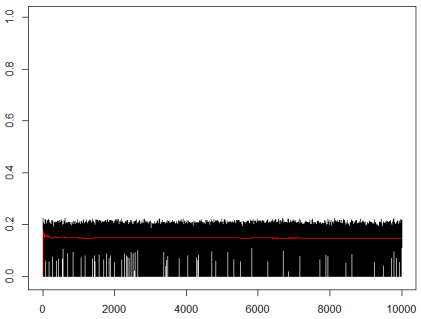


(a) $p(0|y)$ (b) $p(1|y)$

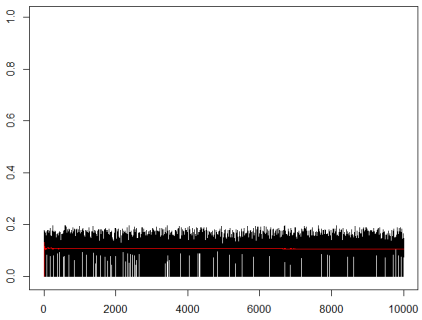
Figura A.1: Mistura di Poisson: esposizione nella scala originale.



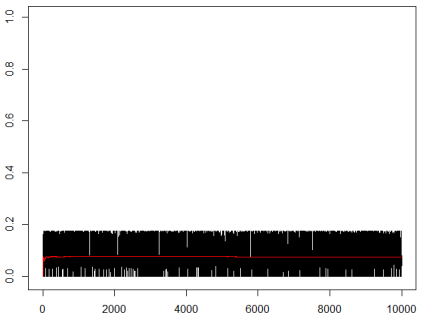
(a) $p(2|y)$



(b) $p(3|y)$

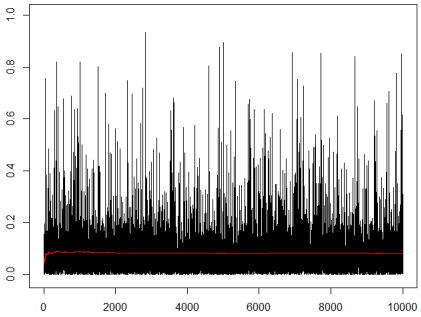


(c) $p(4|y)$

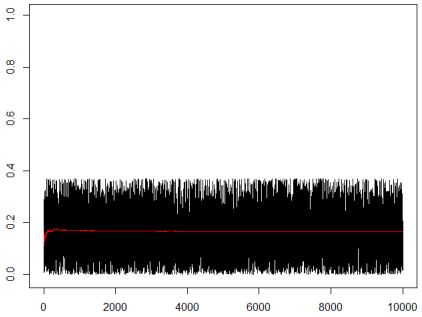


(d) $p(5|y)$

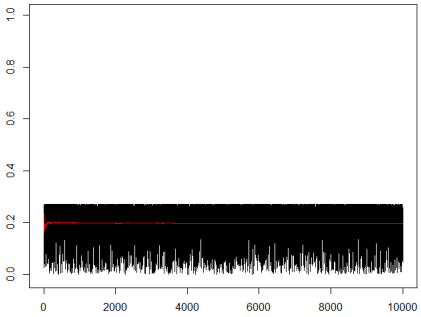
Figura A.2: Mistura di Poisson: esposizione nella scala originale.



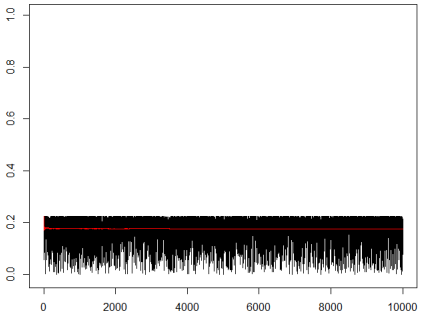
(a) $p(0|y)$



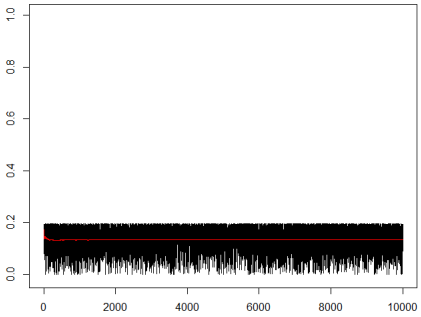
(b) $p(1|y)$



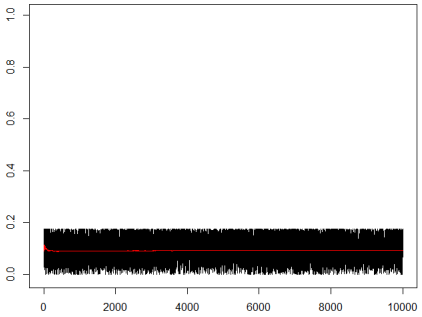
(c) $p(2|y)$



(d) $p(3|y)$

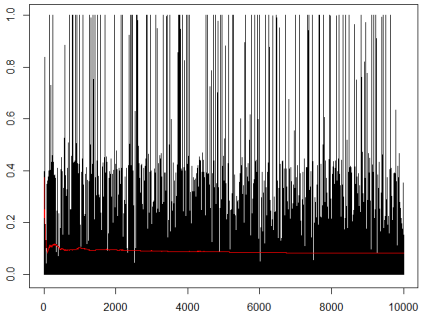


(e) $p(4|y)$

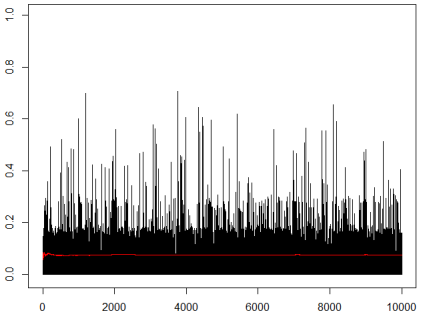


(f) $p(5|y)$

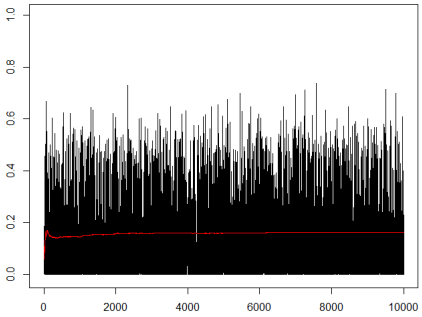
Figura A.3: Mistura di Poisson: esposizione normalizzata.



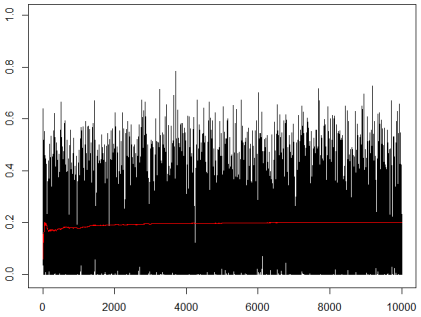
(a) $p(0|y)$



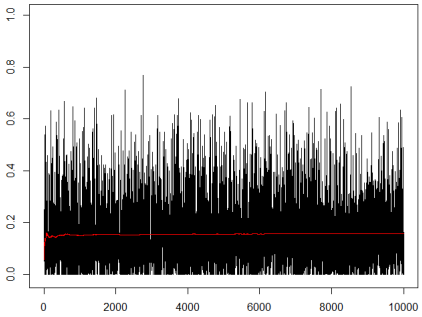
(b) $p(1|y)$



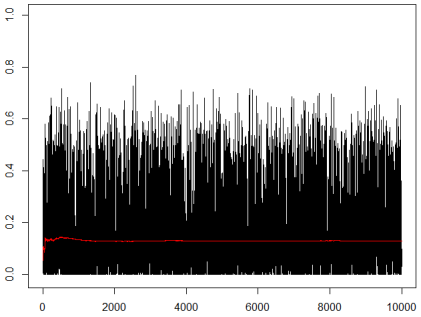
(c) $p(2|y)$



(d) $p(3|y)$

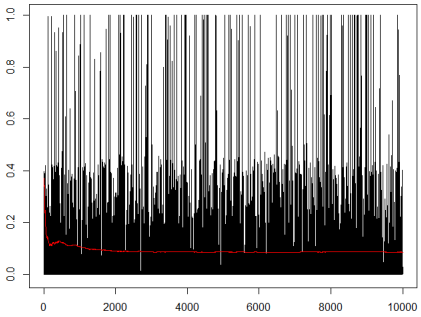


(e) $p(4|y)$

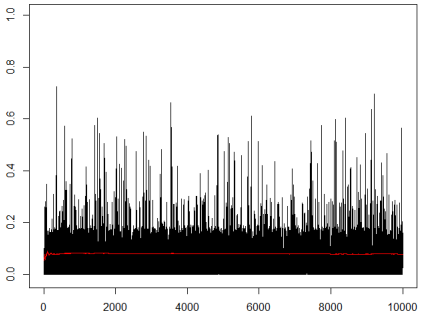


(f) $p(5|y)$

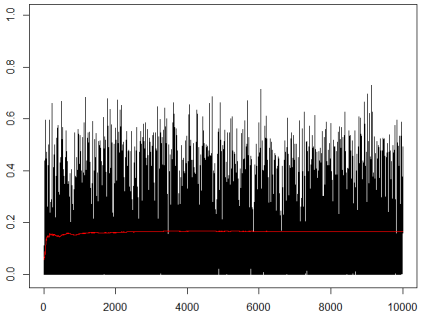
Figura A.4: Mistura di RG: esposizione nella scala originale.



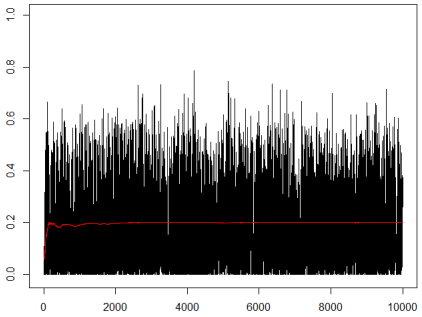
(a) $p(0|y)$



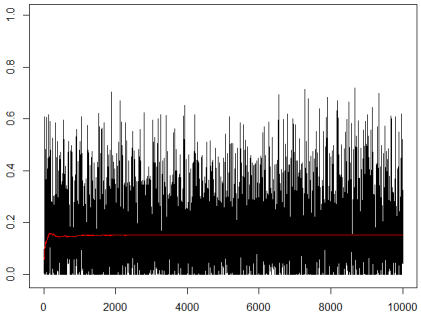
(b) $p(1|y)$



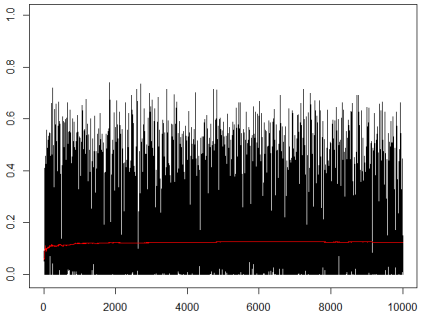
(c) $p(2|y)$



(d) $p(3|y)$



(e) $p(4|y)$



(f) $p(5|y)$

Figura A.5: Mistura di RG: esposizione normalizzata.

Appendice B

Codici R

Pólya urn Gibbs sampler - Rounded Gaussian

```
noGibbs <- function(Nsim, y, K, mu.start, tau.start, theta, sigma,
                    mu0, kappa, alpha, beta)
{
  n <- length(y)
  a <- c(-100,0:(max(y)+1)) # soglie (a0,a1,...)
  S <- matrix(NA, Nsim+1, n)
  S[1,] <- rep(1,n)

  mu.out <- matrix(0, Nsim+1, K)
  tau.out <- matrix(0, Nsim+1, K)
  mu.out[1,1] <- mu.start
  tau.out[1,1] <- tau.start

  for(i in 2:(Nsim+1))
  {
    ### genero y* ###
    us <- runif(n, pnorm(a[y+1], mu.out[(i-1),S[i-1,]],
                         sqrt(1/tau.out[(i-1),S[i-1,]])),
                   pnorm(a[y+2], mu.out[(i-1),S[i-1,]],
                         sqrt(1/tau.out[(i-1),S[i-1,]])))
    ys <- qnorm(us, mu.out[(i-1),S[i-1,]],
                sqrt(1/tau.out[(i-1),S[i-1,]]))


    ## aggiorno allocazione cluster ##
    S[i,] <- S[i-1,]
    for(j in 1:n)
    {
      k_i <- length(unique(S[i,-j]))
      mu.temp <- mu.out[i-1, sort(unique(S[i,-j]))]
      tau.temp <- tau.out[i-1, sort(unique(S[i,-j]))]

      S[i,-j] <- sapply(S[i,-j], function(x) x <- rank(unique(S[i,-j]))
                        [unique(S[i,-j])==x] )

      nuovo.k <- max(S[i,-j])+1
      n_h <- sapply(1:k_i, function(x) sum(S[i,-j]==x))
```

```
      prob <- rep(0, nuovo.k)
      prob[1:k_i] <- (n_h - sigma) * dnorm(ys[j], mu.temp, sqrt(1/tau.temp))
      tau.s <- rgamma(1, alpha, beta)
      mu.s <- rnorm(1, mu0, sqrt(kappa/tau.s))
      prob[nuovo.k] <- (theta + k_i * sigma) * dnorm(ys[j], mu.s,
                                                     sqrt(1/tau.s))
      S[i,j] <- sample(1:nuovo.k, 1, prob=prob)
    }

    ## aggiorno (mu,tau) ##
    n_h <- sapply(1:K, function(x) sum(S[i,]==x))
    ymean_h <- tapply(ys, factor(S[i,], levels=1:K), mean)
    ymean_h[is.na(ymean_h)] <- 0
    yvar_h <- tapply(ys^2, factor(S[i,], levels=1:K), sum) - ymean_h^2*n_h
    yvar_h[is.na(yvar_h)] <- 0

    alpha.s <- alpha + n_h/2
    beta.s <- beta + 0.5 * yvar_h +
                0.5 * n_h/(1+kappa*n_h) * (ymean_h-mu0)^2
    kappa.s <- (1/kappa + n_h)^(-1)
    mu.s <- kappa.s*(1/kappa * mu0 + n_h * ymean_h)

    tau.out[i,n_h!=0] <- rgamma(sum(n_h!=0), alpha.s[n_h!=0],
                                beta.s[n_h!=0])
    mu.out[i,n_h!=0] <- rnorm(sum(n_h!=0), mu.s[n_h!=0],
                              sqrt(kappa.s[n_h!=0]* 1/tau.out[i,n_h!=0]))

    if(i%%500==0) print(i)
  }
  list(mu=mu.out, tau=tau.out, k=S, sigma=sigma, theta=theta)
}
```

Pólya urn Gibbs sampler - Poisson

```
noGibbsPoisson <- function(Nsim, y, K, lambda.start, theta, sigma, a, b)
{
  n <- length(y)
  S <- matrix(NA, Nsim+1, n)
  S[1,] <- rep(1,n)
  lambda.out <- matrix(0, Nsim+1, K)
  lambda.out[1,1] <- lambda.start

  for(i in 2:(Nsim+1))
  {
    ## aggiorno allocazione cluster ##
    S[i,] <- S[i-1,]
    for(j in 1:n)
    {
      k_i <- length(unique(S[i,-j]))
      lambda.temp <- lambda.out[i-1, sort(unique(S[i,-j]))]

      S[i,-j] <- sapply( S[i,-j],
                         function(x) x <- rank(unique(S[i,-j]))
                          [unique(S[i,-j])==x] )

      nuovo.k <- max(S[i,-j])+1
      n_h <- sapply(1:k_i, function(x) sum(S[i,-j]==x))

      prob <- rep(0, nuovo.k)
```

```
      prob[1:k_i] <- (n_h - sigma) * dpois(y[j], lambda.temp)
      lambda.s <- rgamma(1, a, b)
      prob[nuovo.k] <- (theta + k_i * sigma) * dpois(y[j], lambda.s)
      S[i,j] <- sample(1:nuovo.k, 1, prob=prob)
    }

    ## aggiorno lambda ##
    n_h <- sapply(1:K, function(x) sum(S[i,]==x))
    ysum_h <- tapply(y, factor(S[i,], levels=1:K), sum)
    ysum_h[is.na(ysum_h)] <- 0
    a.s <- a + ysum_h
    b.s <- b + n_h

    lambda.out[i,n_h!=0] <- rgamma(sum(n_h!=0), a.s[n_h!=0], b.s[n_h!=0])

    if(i%%500==0) print(i)
  }
  list(lambda=lambda.out, k=S, sigma=sigma, theta=theta)
}
```

Slice Gibbs sampler - Rounded Gaussian

```
sliceGibbsRG <- function(Nsim, K, y, mu.start, tau.start, theta, sigma,
                         mu0, kappa, alpha, beta)
{
  n <- length(y)
  a <- c(-Inf,0:(max(y)+1))

  v <- sapply(1:K, function(x) rbeta(1, 1-sigma, theta+as.numeric(x)*sigma))
  v1 <- 1-v

  pi <- rep(NA,K)
  pi[1] <- v[1]
  pi[2:K] <- sapply(2:K, function(x) v[x]*prod(v1[1:(x-1)]))

  k <- rep(1,n)
  u <- runif(n,0,pi[k])

  mu.out <- matrix(NA, Nsim+1, K)
  tau.out <- matrix(NA, Nsim+1, K)
  pi.out <- matrix(NA, Nsim+1, K)
  k.out <- matrix(NA, Nsim+1, n)

  mu.out[1,] <- mu.start
  tau.out[1,] <- tau.start
  pi.out[1,] <- pi
  k.out[1,] <- k

  for(i in 2:(Nsim+1))
  {
    ### genero y* ###
    us <- runif(n, pnorm(a[y+1], mu.out[(i-1),k], sqrt(1/tau.out[(i-1),k])),
                pnorm(a[y+2], mu.out[(i-1),k], sqrt(1/tau.out[(i-1),k])))
    ys <- qnorm(us, mu.out[(i-1),k], sqrt(1/tau.out[(i-1),k]))

    ## slice ##
    u <- runif(n, 0, pi[k])

    ## aggiorno allocazione cluster ##
```

```
    A.u <- sapply(u, function(x) which(x<pi))
    prob <- t(apply(cbind(y,1:n), 1, function(x)
                p.fun(y=x[1], A.u_index=A.u[[x[2]]], a, mu.out[(i-1),],
                      tau.out[(i-1),], K=K)))

    k <- apply(prob, 1, function(x) sample(1:K, 1, prob=x))
    k.out[i,] <- k

    ## aggiorno pi##
    v <- rep(NA,K)
    for(h in 1:K) v[h] <- rbeta(1, 1-sigma+sum(k==h),
                                theta+h*sigma+sum(k>h))
    v1 <- 1-v

    pi <- rep(NA,K)
    pi[1] <- v[1]
    for(h in 2:K)   pi[h] <- v[h]*prod(v1[1:(h-1)])
    pi.out[i,] <- pi

    ## aggiorno (mu,tau) ##
    n_h <- sapply(1:K, function(x) sum(k==x))
    ymean_h <- tapply(ys, factor(k, levels=1:K), mean)
    ymean_h[is.na(ymean_h)] <- 0
    yvar_h <- tapply(ys^2, factor(k, levels=1:K), sum) - ymean_h^2*n_h
    yvar_h[is.na(yvar_h)] <- 0

    alpha.s <- alpha + n_h/2
    beta.s <- beta + 0.5 * ( yvar_h + n_h/(1+kappa*n_h) * (ymean_h-mu0)^2 )
    kappa.s <- (1/kappa + n_h)^(-1)
    mu.s <- kappa.s*(1/kappa * mu0 + n_h * ymean_h)

    tau.out[i,] <- rgamma(K, alpha.s, beta.s)
    mu.out[i,] <- rnorm(K, mu.s, sqrt(kappa.s * 1/tau.out[i,]))

    if(i%%500==0) print(i)
  }
  list(mu=mu.out, tau=tau.out, pi=pi.out, k=k.out, sigma=sigma, theta=theta)
}

p.fun <- function(y, A.u_index, a, mu, tau, K)
{
  prob <- rep(0,K)
  for(h in A.u_index)
  {
    prob[h] <- pnorm(a[y+2],mu[h],sqrt(1/tau[h])) -
      pnorm(a[y+1],mu[h],sqrt(1/tau[h]))
  }
  prob
}
```

Slice Gibbs sampler - Poisson

```
sliceGibbsPoisson <- function(Nsim, K, y, lambda.start, theta, sigma, a, b)
{
  n <- length(y)
  v <- sapply(1:K, function(x) rbeta(1, 1-sigma, theta+as.numeric(x)*sigma))
  v1 <- 1-v

  pi <- rep(NA,K)
```

```
  pi[1] <- v[1]
  pi[2:K] <- sapply(2:K, function(x) v[x]*prod(v1[1:(x-1)]))

  k <- rep(1,n)
  u <- runif(n,0,pi[k])

  lambda.out <- matrix(NA, Nsim+1, K)
  pi.out <- matrix(NA, Nsim+1, K)
  k.out <- matrix(NA, Nsim+1, n)
  lambda.out[1,] <- lambda.start
  pi.out[1,] <- pi
  k.out[1,] <- k

  for(i in 2:(Nsim+1))
  {
    ## slice ##
    u <- runif(n, 0, pi[k])

    ## aggiorno allocazione cluster ##
    A.u <- sapply(u, function(x) which(x<pi))
    prob <- t(apply(cbind(y,1:n), 1,
                    function(x) p.fun(y=x[1], A.u_index=A.u[[x[2]]],
                                      lambda=lambda.out[(i-1),], K=K)))

    k <- apply(prob, 1, function(x) sample(1:K, 1, prob=x))
    k.out[i,] <- k

    ## aggiorno pi ##
    v <- rep(NA,K)
    for(h in 1:K)   v[h] <- rbeta(1, 1-sigma+sum(k==h),
                                  theta+h*sigma+sum(k>h))
    v1 <- 1-v

    pi <- rep(NA,K)
    pi[1] <- v[1]
    for(h in 2:K)   pi[h] <- v[h]*prod(v1[1:(h-1)])
    pi.out[i,] <- pi

    ## aggiorno (mu,tau) ##
    n_h <- sapply(1:K, function(x) sum(k==x))
    ysum_h <- tapply(y, factor(k, levels=1:K), sum)
    ysum_h[is.na(ysum_h)] <- 0
    a.s <- a + ysum_h
    b.s <- b + n_h

    lambda.out[i,] <- rgamma(K, a.s, b.s)

    if(i%%500==0) print(i)
  }
  list(lambda=lambda.out, pi=pi.out, k=k.out, sigma=sigma, theta=theta)
}

p.fun <- function(y, A.u_index, lambda, K)
{
  prob <- rep(0,K)
  for(h in A.u_index)
  {
    prob[h] <- dpois(y,lambda[h])
  }
  prob
}
```

Bibliografia

Brown, G. & Buckley, W. (2015). Experience rating with Poisson mixtures. *Annals of Actuarial Science*, *9*(2), 304–321.

Canale, A. & Dunson, D. (2011). Bayesian Kernel Mixtures for Counts. *Journal of the American Statistical Association*, *106*, 1528–1539.

Canale, A. & Prünster, I. (2017). Robustifying Bayesian nonparametric mixtures for count data. *Biometrics*, *73*, 174–184.

Cifarelli, D. & Muliere, P. (1989). *Statistica Bayesiana*. Gianni Iuculano.

Clarke, A. & Disney, R. (1985). *Probability and Random Processes: A First Course with Applications*. Wiley.

Dey, D. & Rao, C. (2005). *Bayesian Thinking, Modeling and Computation*. Handbook of Statistics. Elsevier Science.

Ferguson, T. (1973). A Bayesian Analysis of Some Nonparametric Problems. *The Annals of Statistics*, *1*, 209–230.

Ferguson, T. (1974). Prior Distributions on Spaces of Probability Measures. *The Annals of Statistics*, *2*, 615–629.

Ghosal, S., Ghosh, J. & Ramamoorthi, R. (1999). Posterior Consistency of Dirichlet Mixtures in Density Estimation. *The Annals of Statistics*, *27*, 143–158.

Ghosh, J. & Ramamoorthi, R. (2003). *Bayesian Nonparametrics*. Springer.

Haastrup, S. (2000). Comparison of Some Bayesian Analyses of Heterogeneity in Group Life Insurance. *Scandinavian Actuarial Journal*, (1), 2–16.

Hjort, N., Holmes, C., Müller, P. & Walker, S. (2010). *Bayesian Nonparametrics*. Cambridge University Press.

Hougaard, P., Lee, M. & Whitmore, G. (1997). Analysis of Overdispersed Count Data by Mixtures of Poisson Variables and Poisson Processes. *Biometrics*, *53*, 1225–1238.

Ishwaran, H. & James, L. (2001). Gibbs Sampling Methods for Stick-Breaking Priors. *Journal of the American Statistical Association*, *96*, 161–173.

Kalli, M., Griffin, J. & Walker, S. (2011). Slice Sampling Mixture Models. *Statistics and Computing*, *21*, 93–105.

Koenker, R., Gu, J. & I, M. (2017). REBayes. `url:`https://CRAN.R-project.org/package=REBayes.

MacEachern, S. & Müller, P. (1998). Estimating Mixture of Dirichlet Process Models. *Journal of Computational and Graphical Statistics*, *7*, 223–238.

Miller, J. & Harrison, M. (2014). Inconsistency of Pitman-Yor Process Mixtures for the Number of Components. *Journal of Machine Learning Research*, *15*(1), 3333–3370.

Müller, P., Quintana, F., Jara, A. & Hanson, T. (2015). *Bayesian Nonparametric Data Analysis*. Springer.

Neal, R. (2000). Markov Chain Sampling Methods for Dirichlet Process Mixture Models. *Journal of Computational and Graphical Statistics*, *9*, 249–265.

Norberg, R. (1989). Experience Rating in Group Life Insurance. *Scandinavian Actuarial Journal*, (4), 194–224.

Schwartz, L. (1965). On Bayes procedures. *Probability Theory and Related Fields*, *4*, 10–26.

Sethuraman, J. (1994). A Constructive Definition of the Dirichlet Prior. *Statistica Sinica*, *4*, 639–650.

Walker, S. (2007). Sampling the Dirichlet Mixture Model with slices. *Communications in Statistics: Simulation and Computation*, *36*, 45–54.

Ringraziamenti

Questa tesi è stata fin dall'inizio una sfida: sia con me stessa, sia con coloro che hanno sempre criticato la scelta di fare ciò che piace anziché ciò che è utile. Nonostante i momenti di crisi e gli sbalzi d'umore che mi ha comportato, questa tesi è stata una bellissima conclusione per il mio percorso universitario, è stata un'opportunità per imparare cose che altrimenti non avrei mai conosciuto e per ritrovare la curiosità nello studio e il piacere dello studio fine a sé stesso.

Un ringraziamento sincero va quindi al mio relatore, Antonio Canale, per la disponibilità dimostrata; per avermi sempre aiutata, e sopportata, in questi mesi di lavoro sulla tesi.

Ringrazio la mia coinquilina Laura, per avermi spronata e aver creduto in me, e perché nessuno ha mai fatto apparire così interessanti le storie sulle giornate passate in università.

Ringrazio gli amici dell'università: Igor, per i consigli e le straordinarie perle di insolita saggezza; Daniel, che dal primo giorno, senza penna, ha sempre rallegrato le pause con aneddoti e storie di vita; Jacopo e Francesco, per i consigli computazionali, senza i quali probabilmente sarei ancora alle prese con le simulazioni. L'elenco degli amici di Statistica da ringraziare sarebbe infinitamente lungo, ringrazio quindi tutti coloro che in un modo o nell'altro mi hanno sopportata in questi cinque anni, che mi hanno tenuto compagnia per una pausa caffè o un prosecco dopo lezione.