UNIVERSITA' DEGLI STUDI DI MILANO-BICOCCA

Scuola di Economia e Statistica

Corso di Laurea Magistrale in

Scienze Statistiche ed Economiche

DIRICHLET PROCESS

and

HIERARCHICAL MODELS

Tesi di Laurea di:
Alice GIAMPINO
Matricola 790347

Relatore:
Prof.ssa Sonia MIGLIORATI
Correlatore:
Prof. Andrea ONGARO

Anno Accademico 2018 - 2019

Prefazione

Questa tesi scaturisce dalla curiosità di approfondire un argomento tanto in voga al momento quanto misterioso e complicato, traendo quante più nuove conoscenze possibili.

Siccome il mondo ha sempre più bisogno di modi con cui analizzare e modellizzare i dati, i metodi che verranno trattati oltre ad essere attualmente argomenti di ricerca sono anche dei buoni strumenti per "far parlare i dati" e sfruttano tra l'altro uno dei teoremi che più mi affascinano, il *Teorema di Bayes*.

Ho voluto, inoltre, mettere in gioco le conoscenze apprese vedendo con dati reali cosa potesse accadere, se effettivamente i metodi fossero efficienti.

Inoltre, una delle mie passioni è studiare nuovi argomenti che mi permettano di affrontare ogni tematica e che mi diano sempre più strumenti per potermi destreggiare nel mondo dei dati reali, e perché no, avere una tematica di ricerca per un possibile dottorato.

Non sarei riuscita ad arrivare fino a questo livello senza avere il supporto dei miei genitori, di tutte le persone che mi vogliono bene, della persona che mi ama e dei professori che hanno sempre creduto in me. Sono state fondamentali tutte le persone che ho conosciuto in questo mio percorso, tutti coloro che con pazienza e consigli mi hanno guidata attraverso questi anni. Grazie a tutti per il vostro supporto.

Indice

1
Introduzione

1.1
Obiettivi e tematiche trattate

Negli ultimi anni si è approfondito sempre più lo studio della statistica bayesiana e in particolare della statistica bayesiana non parametrica. Quest'ultima ha visto un'applicazione costantemente più ampia in diversi ambiti quali classificazione, genetica, econometria, apprendimento automatico e altri ancora.

La statistica bayesiana non parametrica permette, in alcune casistiche, di ottenere performance migliori rispetto ad approcci parametrici. E' una classe di modelli che hanno potenzialmente un numero infinito di parametri, consentendo così maggior flessibilità quando, appunto, il numero di parametri aumenta al crescere dei dati.

E' inoltre molto utilizzata anche nel caso di *clustering*, poichè invece di fissare il numero di gruppi lo si lascia fissare dai dati. Il numero aumenterà con la quantità di dati a disposizione, ovviamente introducendo una penalizzazione tramite un parametro che ridurrà così il rischio di overfitting (cluster non parametrici)[36].

Questa tesi ha il fine di approfondire uno dei modelli più comunemente utilizzati nella statistica bayesiana non parametrica, il cosiddetto *Dirichlet Process*. Tale processo è usato spesso come base dei modelli probabilistici bayesiani non paramatrici.

Ci occuperemo, inizialmente, di presentare il concetto di scambiabilità con l'obbiettivo di illustrare il processo di Dirichlet e le sue proprietà.

Successivamente descriveremo dei modelli bayesiani non parametrici che si basano su tale metodologia e che ne permettono un'ampia applicazione.

Nel quarto capitolo proseguiremo illustrando processi bayesiani che vanno oltre il *Dirichlet process*, tra i quali il ristorante cinese e il franchise di ristoranti cinesi, molto utili nell'analisi dei gruppi.

Infine, presenteremo un esempio di applicazione dei modelli bayesiani non parametrici nell'analisi testuale e proporremo un'applicazione su dati raccolti da Twitter utilizzando uno specifico modello per questa tipologia di analisi.

Nella parte finale dell'elaborato discuteremo di possibili utilizzi futuri di questa metodologia.

1.2 Terminologia

Con **modello statistico** su uno spazio campionario **X**, intendiamo un insieme di misure di probabilità su **X**.

Se **PM(X)** è lo spazio di tutte le misure di probabilità su **X**, allora, un modello M è un sottoinsieme t.c. $M \subset$ **PM(X)**. L'assunzione di fondo è che tale spazio sia misurabile. Gli elementi di M sono indicizzati dal parametro θ che assume valori nello **spazio parametrico** Θ,

$$M = \{P_\theta | \theta \in \Theta\}, \tag{1.1}$$

dove P_θ è un elemento di **PM(X)**.

Chiamiamo un modello **parametrico** se Θ ha dimensioni finite, se invece ha dimensione infinite, M è in questo caso un **modello nonparametrico**[27].

Focus di questa tesi sarà trattare modelli **nonparametrici**, ma in ambito bayesiano.

Teorema fondamentale è il teorema di Bayes.

Teorema 1 (Bayes)**.** *Siano A e B due eventi, e sia B possibile, cioè richiediamo che* $P(B) \neq 0$*. La probabilità a posteriori di A condizionato a B può essere calcolata nel modo seguente:*

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)} \tag{1.2}$$

Un modello bayesiano consiste in un modello M, inerente alle osservazioni campionarie, e una prior H[27]. In questo caso, abbiamo che:

$$\begin{aligned} \theta &\sim H \\ X_1, X_2, \ldots | \theta &\sim_{iid} P_\theta \end{aligned} \tag{1.3}$$

dove $X_1, X_2, \ldots$ rappresentano le variabili casuali relative al campione ottenuto da **X**. E' importante sottolineare che in questo modo stiamo indicando non variabili $i.i.d$, bensì variabili *condizionatamente* $i.i.d$ e che la distribuzione di θ è detta **prior distribution** (o semplicemente **prior**). L'aver definito in questo modo le due distribuzioni consente di ottenere quello che è l'obiettivo di questa tipologia di analisi, cioè la **posterior distribution** (o **posterior**):

$$\Pi(\theta \in \Theta | X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n) \tag{1.4}$$

La statistica bayesiana utilizza questa distribuzione per esprimere l'incertezza della stima parametrica basata sul campione osservato.

L'approccio non parametrico, invece, non assume una distribuzione alla quale vincolare la stima dei parametri i quali hanno uno spazio *infinito* dimensionale.

La statistica bayesiana non parametrica riguarda modelli e metodi caratterizzati da:

(a) un “grande” spazio parametrico;

(b) costruzione di misure di probabilità su questo spazio [19].

Un **modello bayesiano non parametrico**, quindi, è un modello bayesiano i cui parametri hanno uno spazio infinito dimensionale. Per caratterizzarlo, bisogna caratterizzare una distribuzione di probabilità (prior) definita su tale spazio.

Dobbiamo, per questo motivo, definire una distribuzione sullo spazio Θ infinito dimensionale, ovvero un processo stocastico che abbia traiettorie in Θ. La statistica bayesiana non parametrica è una semplice costruzione di funzioni di densità casuali senza alcuna restrizione di forma che sfrutta i processi stocastici. I più utilizzati sono i processi Gaussiani e i processi ad incrementi indipendenti.

Il processo stocastico è governato dalla prior, tra le più comuni vi è il **processo di Dirichlet** [12] che ha una traiettoria campionaria che si comporta quasi certamente come una funzione di distribuzione discreta. Spesso vengono utilizzati modelli di misture costruiti dal processo di Dirichlet, come misture di distribuzioni che generano funzioni di densità casuali [23].

2
Dirichlet Process

Prima di parlare del processo di Dirichlet è utile andare a trattare uno dei teoremi fondamentali per la sua applicazione.

2.1
Scambiabilità e Teorema di de Finetti

Quando abbiamo osservazioni $i.i.d$ l'ordine con cui vengono osservate non dovrebbe essere importante, per questo si parla di modelli di apprendimento **scambiabili**.

Definizione 1. *La sequenza $(X_n)_n \geq 1$ è scambiabile se per ogni $n \geq 1$ e ogni permutazione π di (1, 2, . . . , n) [1]*

$$(X_1, X_2, \ldots, X_n) \stackrel{\mathrm{d}}{=} (X_{\pi(1)}, X_{\pi(2)}, \ldots, X_{\pi(n)} \tag{2.1}$$

.

E' un'assunzione debole e quando non si hanno abbastanza informazioni si ricorre a questa condizione di simmetria. La Definizione 1 può essere espressa come:

Definizione 2. *Sia $\boldsymbol{x} = (x_1, x_2, x_3, \ldots)$ una successione infinita numerabile di valori reali, e $\boldsymbol{x}_k = (x_1, x_2, \ldots, x_k)$ per un certo $k \in N$. Se la misura di probabilità per $\boldsymbol{x}_k$ è invariante sotto permutazioni degli elementi di $\boldsymbol{x}_k$, ossia se*

$$p(x_1, x_2, ..., x_k) = p(x_{\pi(1)}, x_{(2)}, \ldots, x_{\pi(k)}) \tag{2.2}$$

per ogni permutazione π dell'insieme $1, \ldots, k$, diremo che $\boldsymbol{x}_k$ è scambiabile, dove p è la densità rispetto alla misura di probabilità. Se $\boldsymbol{x}_k$ è scambiabile per ogni k, diremo che anche $\boldsymbol{x}$ è scambiabile. Se $\boldsymbol{x}_k$ può essere inclusa in una successione scambiabile $\boldsymbol{x}$, diremo che $\boldsymbol{x}_k$ è estendibile e scambiabile all'infinito[8].

Il teorema di de Finetti rappresenta il fondamento nell'interpretazione della teoria delle probabilità. De Finetti diede risultati sull'interscambiabilità, usando il termine "equivalenza" per sequenze di prove su un dato fenomeno, tutte fatte in condizioni analoghe

Teorema 2 (Rappresentazione di de Finetti.)**.** *Se $\boldsymbol{x}$ è scambiabile e la sua funzione di ripartizione empirica è descritta tramite un parametro θ, allora la probabilità di ottenere una sequenza $\boldsymbol{x}_k$ è data da*

$$p(\boldsymbol{x}_k) = \int \prod_{i=1}^{k} p(x_i|\theta)\, dH(\theta) \tag{2.3}$$

Il teorema fu inizialmente specificato per valori di x_i in $\{0,1\}$ implicando così che il modello è *Bernoulli* e la prior H. La sua applicazione comporta la costruzione della prior appartenente ad un opportuno insieme di funzioni di densità e vi è un meccanismo di aggiornamento che tiene in considerazione i valori osservati.

Se ci restringessimo al caso in cui ogni variabile casuale presente nella successione possa assumere n valori discreti, allora la prior che ne deriva è $H(\mathbf{p})$ e descrive lo stato rispetto a $\mathbf{p}$, sulla quale poi vanno mediate le funzioni di probabilità p_j, dove $\mathbf{p}$ è un vettore di probabilità ignota,

$$p(\mathbf{x}_k) = p(x_1, x_2, \ldots, x_k) = \int_{S_n} \prod_{i=1}^{k} \; p(x_i) \; H(\mathbf{p}) \; d\mathbf{p}$$

e

$$S_n = \left\{ \; \mathbf{p} \; : \; p_j \geq 0, \forall j \quad : \quad \sum_{j=1}^{n} p_j = 1 \right\}$$

è lo spazio di probabilità relativo, dove $p(x_j) = p_j$ e $p(x_1, x_2, \ldots, x_k)$ è la funzione di probabilità congiunta.

Possiamo affermare che il teorema di de Finetti ci garantisce il collegamento e l'equivalenza logica fra l'approccio frequentista a livello operativo e l'ambito della teoria bayesiana. Vale anche il viceversa, ovvero che un certo valore per il parametro θ^* non noto in un approccio frequentista equivale ad una prior $H(\theta) = \delta(\theta - \theta^*)$ che permette di ottenere esattamente ciò che ci si aspetta in un modello frequentista, cioè

$$p(\mathbf{x}_k) = \prod_{i=1}^{k} p(x_i | \theta = \theta^*)$$

L'assunzione di scambiabilità e il teorema di de Finetti sono alla base della deduzione della teoria delle probabilità e, in particolare, il teorema ci indica anche un altro aspetto di cui tener conto: se in una successione di variabili casuali non conta l'ordine degli elementi, allora qualsiasi probabilità $p(\mathbf{x}_k)$ può essere generata attraverso l'equazione (2.3) a partire da una prior $H(\theta)$.

Questo aspetto non sarà ulteriormente approfondito in questa tesi, ma si può consultare il Capitolo 18 di [21].

2.2 Dirichlet Process

Il *processo di Dirichlet* (DP, *Dirichlet Process*) è un processo stocastico molto utilizzato nei modelli bayesiani non parametrici. In particolare, ne deriva una classe di distribuzioni di probabilità largamente usata come prior nei modelli mistura infiniti, cioè modelli mistura basati sul processo di Dirichlet. Possiamo considerarli come distribuzioni su spazi di funzione, cioè, nel caso di DP, si tratta di distribuzioni su misure di probabilità che sono interpretabili come distribuzioni su un qualche spazio probabilistico. Ecco

perché si dice che il processo di Dirichlet è una "distribuzione di distribuzioni". Le distribuzioni estratte da questo processo sono discrete, ma non sono rappresentabili attraverso un numero finito di parametri, il modello che ne deriva è quindi non parametrico.

Il processo di Dirichlet venne formalizzato da Ferguson nel 1973 e il nome deriva dal fatto che le sue distribuzioni finito-marginali sono distribuite come una variabile casuale di Dirichlet.

2.2.1 Definizione di Processo di Dirichlet

Sia $(\Theta, \mathscr{B}, P)$ uno spazio di probabilità, in cui Θ è lo spazio campionario, $\mathscr{B}$ la σ-algebra di Borel dei sottoinsiemi di Θ e P una misura di probabilità. Sia H una distribuzione di probabilità su tale spazio, e α un numero reale positivo. Allora G si distribuisce secondo un processo di Dirichlet se, per ogni partizione finita e misurabile $(A_1, A_2, \ldots, A_r)$ di Θ, il vettore casuale $(G(A_1), G(A_2), \ldots, G(A_r))$ è distribuito come una variabile casuale di Dirichlet con parametri di distribuzione $(\alpha H(A_1), \alpha H(A_2), \ldots, \alpha H(A_r))$, ovvero

$$(G(A_1), G(A_2), \ldots, G(A_r)) \sim Dir(\alpha H(A_1), \alpha H(A_2), \ldots, \alpha H(A_r)) \tag{2.4}$$

dove *Dir( )* indica la distribuzione di Dirichlet (Appendice A.6). Possiamo scrivere che $G \sim DP(\alpha, H)$.

La distribuzione di Dirichlet si basa sulla misura di Lebesgue, cioè la misura solitamente utilizzata per i sottoinsiemi di uno spazio euclideo di dimensione n. Gli insiemi a cui è possibile assegnare una misura di Lebesgue sono detti misurabili secondo Lebesgue o Lebesgue-misurabili [6].

Nella definizione del DP, H è detta *distribuzione di base* ed è la media del DP, mentre α è il *parametro di concentrazione* e può essere pensato come una varianza inversa.

Grazie alle proprietà della distribuzione e alla definizione di DP, abbiamo che:

$$G(A) \sim Beta(\alpha H(A), \alpha(1 - H(A)), \qquad \forall A \in \mathscr{B} \tag{2.5}$$

I momenti di questa distribuzione risultano essere:

$$\begin{aligned} E[G(A)] &= H(A) \\ Var[G(A)] &= \frac{H(A)(1 - H(A))}{(\alpha + 1)} \end{aligned} \tag{2.6}$$

Guardando con occhio più critico alla valenza dei parametri della distribuzione, possiamo notare che all'aumentare di α, diminuisce la varianza, cioè il processo avrà una concentrazione maggiore nei valori prossimi alla media, per questo viene detto *parametro di precisione*. Invece H, detto anche *parametro di forza*, indica con quanta "forza" entra la priori con distribuzione DP in un modello Bayesiano non parametrico.

Un'osservazione va fatta riguardo il primo parametro, ovvero per $\alpha \to \infty$ abbiamo che $G(A) \to H(A)$ per ogni misurabile A. Quindi, $G \to H$ puntualmente. Le distribuzioni estratte da un DP risultano essere di tipo discreto

con distribuzione di probabilità che somma a uno, anche se H è continua. Perciò le due non hanno bisogno di essere assolutamente continue l'una rispetto all'altra.

2.2.2 Costruzione del Processo di Dirichlet

Analizziamo ora come poter costruire il processo di Dirichlet, prima in modo "naive" per capirne il meccanismo e poi in modo più consono.

Siccome le distribuzioni congiunte sono specificate in modo consistente, allora possiamo vedere G come una funzione della σ-algebra di Borel $\mathscr{B}$ sull'intervallo unitario. Abbiamo, quindi, la capacità di costruire le distribuzioni marginali su uno spazio non numerabile che sia il prodotto di singoli spazi, cioè $[0,1]^{\mathscr{B}}$. Tale realizzazione è possibile grazie al teorema di consistenza di Kolmogorov (Appendice B.1) che assicura che una collezione opportunamente "consistente" di distribuzioni a dimensione finita definisce un processo stocastico.

Purtroppo, ci sono due principali problematiche nell'utilizzo di questo approccio. La prima è che il prodotto su $[0,1]^{\mathscr{B}}$ non è abbastanza ampio da contenere lo spazio di misure di probabilità. E' possibile risolvere questo problema lavorando con misure esterne, a condizione che si possa dimostrare che G è additivo numerabile. In questo caso emerge la seconda problematica, cioè per una data sequenza di insiemi disgiunti A_n, è verificato che $G(\bigcup_{n=1}^{\infty} A_n) = \sum_{n=1}^{\infty} G(A_n)$, ma l'insieme nullo dipende dalla sequenza A_n e poiché il numero di tali sequenze non è numerabile, l'approccio "naive" che utilizza il teorema di consistenza di Kolmogorov non riesce a raggiungere il risultato finale.

Partendo da quanto visto finora, è necessario utilizzare uno spazio che sia numerabile, $\mathscr{F}$ per $\mathscr{B}$, tale per cui ogni misura di probabilità P sia una funzione $\mathscr{F} \to [0,1]$. In questo modo si evitano le difficoltà precedenti che non si presentano sul prodotto numerabile $[0,1]^{\mathscr{F}}$.

A questo punto possiamo considerare un altro approccio di costruzione basato sulla normalizzazione. Ovvero, riguarda la normalizzazione di un processo Gamma con misura di intensità α. Un processo Gamma è un processo a incrementi indipendenti che deriva dalla teoria generale del processo di Lévy. Questo processo è utile quando visto come rappresentazione del processo di Dirichlet perché permette di trovare la distribuzione della media di G e la stima delle code quando G segue un DP su $\mathbb{R}$.

2.2.3 Proprietà

Il processo di Dirichlet ha diverse proprietà, legate sia alla sua distribuzione che al suo utilizzo, che vengono immediatamente ottenute una volta costruito il processo.

Partendo dalla distribuzione di G possiamo constatare le seguenti:

1. *Distribuzioni marginali.* Prendendo in considerazione la distribuzione di una variabile casuale $Beta$ (Appendice A.4) e considerando

una partizione $\{A, A^c\}$ allora $G(A) \sim Beta(\alpha H(A), \alpha H(A^c))$, dove α è la misura di intensità. Il cui valore atteso risulta essere

$$E[G(A)] = \frac{\alpha H(A)}{\alpha H(A) + \alpha H(A^c)} = H(A) \qquad H(A) = \frac{\alpha H(A)}{\alpha}$$

Quindi, $H(A)$ è una misura di probabilità e $\theta|H \sim H$, allora la distribuzione marginale di θ è H. Ecco perché H può essere chiamata *misura centrale*.

2. *Funzionali lineari.* Dalla relazione $E[G(A)] = H(A)$ si può estendere la misura standard in modo sequenziale a semplici funzioni misurabili, funzioni misurabili non negative e a tutte le funzioni integrabili. Ciò avviene perché, per la relazione precedentemente scritta, se ψ è una funzione integrabile di H allora $E(\int \psi dG) = \int \psi dH$. Se si volesse si potrebbe ottenere la distribuzione di $\int \psi dG$ anche analiticamente, ma risulterebbe molto più complicato rispetto ad utilizzare la distribuzione $Beta$ di $G(A)$ perché richiederebbe l'utilizzo di meccanismi complicati come presentati nei paper Regazzini, Guglielmi e Di Nunno (2002) [32] e Hjort and Ongaro (2005) [18].

3. *Correlazione negativa.* Si riscontra correlazione negativa tra le probabilità di ogni coppia di insiemi disgiunti. Ci si potrebbe aspettare che ci sia una concentrazione della distribuzione nelle zone circostanti agli intorni tra i due insiemi e che possa aumentare o diminuire per entrambi, ma ciò non avviene poiché il processo di Dirichlet non considera la topologia dello spazio nell'assegnare la concentrazione della distribuzione.

4. *Discretezza.* Una delle proprietà che più caratterizzano il processo di Dirichlet è la discretezza delle distribuzione campionate da esso. Una distribuzione G è discreta se e solo se $G(\theta : G\{\theta\} > 0) = 1$.

   Considerando il modello $\theta|H \sim H$ e $G \sim DP(,H)$, questa proprietà dice che

   $$(\alpha H \times G)\{(G, \theta) : G\{\theta\} > 0\} = 1 \tag{2.7}$$

   Equivalentemente

   $$(G \times (\alpha + \delta_\theta)H)\{(\theta, G) : G\{\theta\} > 0\} = 1 \tag{2.8}$$

   dove H è la distribuzione marginale di θ e $(\alpha + \delta_\theta, H)$ sono i parametri della distribuzione condizionata di $G|\theta$. Il processo di Dirichlet assicura che tutti i campioni casuali dalla posterior del processo assegnino una concentrazione (massa) positiva al punto θ. La discretezza non è un ostacolo alle buone proprietà di convergenza degli stimatori considerando che la distribuzione empirica è anch'essa discreta ma converge uniformemente a qualsiasi vera distribuzione.

5. *Supporto.* Dal processo di Dirichlet possono essere campionate solo distribuzioni discrete, non per questo il supporto di tale processo è necessariamente piccolo.

Il supporto del processo di Dirichlet è composto da tutti i supporti delle singole misure di probabilità G^* tali che quest'ultimi siano contenuti nel supporto della prior H. Risulta quindi:

$$supp(DP) = \{G^* : supp(G^*) \subset supp(H)\} \tag{2.9}$$

Il supporto di G^* nel caso in cui A non appartenesse al supporto di H deve essere tale che $G^*(A) = 0$. Abbiamo che $H(A) = 0$, di conseguenza $G(A) = 0$ quasi certamente nel supporto del DP. D'altro canto, l'approssimazione in forma debole viene mantenuta se le probabilità di una partizione sono approssimate correttamente. Questa proprietà può essere garantita dalla non singolarità della distribuzione di Dirichlet con parametri positivi. Ergo per cui, se H è a supporto pieno, come una distribuzione *Normale* (Appendice A.2), allora ogni misura di probabilità è nel supporto del DP.

6. *Self-similarity.* Riguarda la distribuzione di sottoinsiemi che seguono sempre la distribuzione di Dirichlet, ma con scala differente. Questa è una proprietà che distigue il DP dagli altri processi.

   Quindi, sia A un insieme tale che $0 < H(A) < 1$ il che assicura di avere $0 < G(A) < 1$ per quasi tutti i campioni dal processo di Dirichlet. Definiamo $G|_A$ la restrizione di G su A che è la distribuzione di probabilità definita da $G|_A(B) = \frac{G(A \cap B)}{G(A)}$ e in maniera similare $G|_{A^c}$. Allora, $G|_A$ e $G|_{A^c}$ sono mutualmenti indipendenti e il processo $\{G|_A, G|_{A^c}\}$ è definito, per cui $G|_A$ segue un $DP \sim Dir(\alpha H(A), H|_A)$.

   Per ogni insieme dato A, la concentrazione della distribuzione *within* A è indipendente da quella *within* A^c, ed entrambe sono indipendenti da quella assegnata a tutto l'insieme A. Inoltre, la distribuzione del processo *within* A segue, come già anticipato, un processo di Dirichlet con un'appropriata scala. Questa proprietà ha le sue radici nella relazione tra le variabili $Gamma$ (Appendice A.3) indipendenti e la variabile distribuita come una $Dirichlet$ che si forma dal loro rapporto: se $X_1, \ldots, X_k$ sono variabili $Gamma$ indipendenti, allora $X = \sum_{i=1}^{k} X_i$ e $\left(\frac{X_1}{X}, \ldots, \frac{X_k}{X}\right)$ sono indipendenti.

   Ci sono diverse conseguenze di questa proprietà. Tra cui è importante ricordare che il processo di Dirichlet può essere generato distribuendo la concentrazione in maniera sequenziale su varie sub-regioni seguendo una struttura ad albero. La proprietà *tail-freeness*, cioè l'indipendenza ai vari livelli di allocazione, è utile perché permette di ottenere un supporto grande per la prior e una consistenza debole per la posterior. Di notevole importanza ricordare che il DP è l'unico processo *tail-free* dove la scelta della partizione non è rilevante.

7. *Limiti.* Dobbiamo tenere in considerazione anche i diversi tipi di limite di questo processo. Difatti, quando consideriamo una sequenza di processi di Dirichlet tale che le misure centrali della distribuzione convergono al limite a H, possiamo trovare tre tipi di limite:

(i) se la massa totale va ad infinito, allora la sequenza converge alla prior degenere a H;

(ii) se la massa totale va ad un numero finito α non nullo, allora il limite è $DP(\alpha, H)$;

(iii) se la massa totale va a zero, allora il processo sceglierà un punto random di H e avrà la massa totale concentrata in quel punto.

Data queste convergenze delle misure centrali, la *tightness* è automatica, mentre le distribuzioni finito dimensionali sono distribuzioni di Dirichlet che convergono ad un limite appropriato alla convergenza dei momenti misti.

Questa proprietà ha conseguenze in due scenari differenti: la posterior di Dirichlet converge debolmente ad un bootstrap Bayesiano quando il parametro di precisione va a zero, e converge alla misura degenere a G_0, vera distribuzione, come il campione di ampiezza n tende ad infinito [19]. In questo modo si ha che l'intera posterior di G è debolmente consistente a G_0. Questa convergenza si rafforza automaticamente con la convergenza in distanza di *Kolmogorov-Smirnov* e con il teorema di *Glivenko-Cantelli* per la distribuzione empirica. Più nel dettaglio si possono trovare le due spiegazioni in Appendice B.3 e B.4.

E' utile notare che non è stata posta alcuna condizione sulla prior, quindi la consistenza è presente qualsiasi sia la scelta della prior, anche quando la vera distribuzione non è nel supporto di quest'ultima.

Solitamente avere la vera distribuzione nel supporto della prior è uno dei requisiti minimi affinché si possa avere una posterior consistente. Di conseguenza, questo è molto peculiare in un contesto Bayesiano. Dal momento che, quando con la prior si escludono delle regioni, la posterior, che si ottiene moltiplicando la prior con la funzione di verosimiglianza e normalizzando, dovrebbe escludere le stesse regioni.

Il problema sopraccennato viene risolto col fatto che, in questo caso specifico, non si utilizza il teorema di Bayes (Teorema 1).

8. *Campioni.* Non tratteremo solo i possibili campioni derivanti da un processo di Dirichlet, ma anche i possibili pattern riscontrabili nei dati. Il fatto che da un DP si possano campionare solo distribuzioni discrete comporta a livello di proprietà di discretezza a generare legami tra le osservazioni estremamente utili in clustering.

   Più nello specifico, se consideriamo la distribuzione congiunta marginale delle n osservazioni $(\theta_1, \ldots, \theta_n)$ da G, dove G può essere descritto sequenzialmente ed è un campione da $DP(\alpha, H)$, si ha

$$\theta_2 | G, \theta_1 \sim G \quad e \quad G|\theta_1 \sim DP\left(\alpha+1, \frac{\alpha}{\alpha+1}H + \frac{1}{\alpha+1}\delta_{\theta_1}\right) \tag{2.10}$$

dove $\theta_1 \sim H$ marginalmente.

E' facile notare che se si eliminasse G, rimarrebbe $\theta_2|\theta_1 \sim \frac{\alpha}{\alpha+1}H + \frac{1}{\alpha+1}\delta_{\theta_1}$, cioè la distribuzione di θ_2 dato θ_1 con probabilità $\frac{1}{\alpha+1}$ e si ottiene una nuova estrazione da H con probabilità $\frac{\alpha}{\alpha+1}$.

Quindi andando avanti nella sequenza delle nostre variabili e considerando θ_n date $\theta_1,\ldots,\theta_{n-1}$ allora seguendo il ragionamento soprastante possiamo scrivere che θ_n duplicherà ogni θ_i precedente con probabilità $\frac{\alpha}{\alpha+n-1}$.

Un'altra accortezza che possiamo tener in considerazione è che in $(\theta_1,\ldots,\theta_n)$ possiamo avere delle ripetizioni. Questo fa sì che ci sia un cambio a livello di probabilità. Definiamo $\theta_1^*,\ldots,\theta_k^*$ come i valori distinti delle k distinte osservazioni in $(\theta_1,\ldots,\theta_{n-1})$. Per questo motivo, possiamo considerare θ_j^* come l'estrazione di variabili condizionanti e la probabilità diventa $\frac{n_j}{\alpha+n-1}$. θ_j^* non contiene ripetizioni, quindi ha i solo valori unici di $\{\theta_1,\ldots,\theta_{n-1}\}$ con frequenza n_j rispettivamente $(j = 1,\ldots,k)$. Allora in questo caso l'estrazione da G con probabilità $\frac{\alpha}{\alpha+n-1}$ diventa:

$$\theta_n|\theta_1,\ldots,\theta_{n-1} \sim \begin{cases} \delta_{\theta_j^*}, & \textit{con probabilità} \quad \frac{n_j}{\alpha+n-1} \quad j = 1,\ldots,k \\ H, & \textit{con probabilità} \quad \frac{\alpha}{\alpha+n-1} \end{cases} \tag{2.11}$$

Quando abbiamo la proprietà di scambiabilità tra $(\theta_1,\ldots,\theta_n)$ allora possiamo ricondurci alla procedura trattata da *Blackwell e MacQueen (1973)* [4]. Gli autori si riferiscono allo schema generalizzato di Pólya urn, quest'ultima sarà trattata nel capitolo successivo dove ne mostreremo l'importanza per lo sviluppo del MCMC (*Monte Carlo Markov Chain*) per le variabili latenti campionate da un DP. Quindi, in questo caso, per ogni θ_i dato θ_j con $j = 1,\ldots,i-1,i+1,\ldots,n$ possiamo applicare quanto detto finora.

Il fatto che non si abbiano tutte osservazioni distinte e che siano legate tra loro porta, ovviamente, ad avere un numero totale di estrazioni da H includendo la prima, che è generalmente più piccola di n. Quindi, in modo sequenziale, la probabilità di avere dalle estrazioni una nuova osservazione agli step $1,2,\ldots,n$ è $1,\frac{\alpha}{\alpha+1},\ldots,\frac{\alpha}{\alpha+n-1}$. Di conseguenza, il numero atteso di valori distinti varia col variare di n ed in particolare:

$$E(K_n) = \sum_{i=1}^{n} \frac{\alpha}{\alpha+i-1} \sim \alpha \log\left(\frac{n}{\alpha}\right) \quad \textit{come} \quad n \to \infty \tag{2.12}$$

dove K_n è il numero di valori distinti e si può ottenere la sua distribuzione esatta, nonché le approssimazioni *Normale* e di $Poisson$ (Appendice A.1). La crescita logaritmica di K_n provoca sparsità nei dati e spesso viene utilizzato in applicazioni di machine learning [19].

9. *Code della distribuzione.* La coda di G è molto più stretta rispetto a quanto si possa pensare. Difatti, siccome $E(G) = H$ si è portati a credere che H e G abbiano code in media equivalenti.

   Questo è possibile grazie alla proprietà asintotica che permette matematicamente di avere una coda di queste dimensioni.

   C'è da fare una precisazione nel caso in cui H fosse una normale standard. In questo caso, la coda di $G(X > x)$ è più sottile di $e^{-e^{\frac{x^2}{2}}}$ quasi certamente per tutte le x sufficientemente grandi. Quindi, è più spessa della coda di una gaussiana.

   Perciò se $H \sim Cauchy$ (Appendice A.5) standard allora G ha momenti finiti, nonostante la distribuzione di partenza non abbia nemmeno la media definita.

10. *Singolarità mutuale.* Riguarda un'altra proprietà molto interessante, cioè ogni coppia di processi di Dirichlet è composta da due processi tra loro mutualmente singolari a meno che non condividano lo stesso nucleo.

    Quindi, se la prior è non-atomica, allora la prior e la posterior di un processo di Dirichlet sono mutualmente singolari. Il teorema di Bayes implicherebbe che la posterior deve essere assolutamente continua rispetto alla distribuzione a priori. In questo caso, non si sta applicando il teorema di Bayes.

2.2.4 Distribuzione a posteriori

Siccome uno dei problemi inferenziali non parametrici è quello di stimare una misura di probabilità, è utile rivolgere per un momento l'attenzione al caso parametrico del modello multinomiale che specifica una distribuzione di probabilità arbitraria allo spazio campionario degli interi finiti. Questo modello può derivare da una distribuzione arbitraria ottenuta raggruppando i dati in un numero finito di categorie.

Possiamo considerare $(\pi_1, \ldots, \pi_k)$ le probabilità delle categorie con frequenze $n_1, \ldots, n_k$, allora la funzione di verosimiglianza risulta essere proporzionale a $\pi_1^{n_1}, \ldots, \pi_k^{n_k}$. La forma della verosimiglianza combacia con la forma finito-dimensionale della prior di Dirichlet che, tenendo conto della restrizione $\sum_{i=1}^{k} \pi_i = 1$ è relativa alle $k-1$ componenti, ha densità proporzionale a $\pi_1^{c_1-1}, \ldots, \pi_k^{c_k-1}$ [14].

La posterior, che non è altro che la moltiplicazione tra prior e funzione di verosimiglianza normalizzate, allora sarà proporzionale a $\pi_1^{n_1+c_1-1}, \ldots, \pi_k^{n_k+c_k-1}$, che risulta di nuovo essere una distribuzione di Dirichlet.

Dobbiamo tenere in considerazione la proprietà di coniugatezza che risulterà molto importante. Tratteremo poi nel dettaglio anche la consistenza della posterior.

Consideriamo ora G che, come detto, è una distribuzione casuale da cui possiamo estrarre dei campioni. Allora, possiamo estrarli da G in modo che

siano una sequenza di n campioni indipendenti $\theta_1,\dots,\theta_n$. Ogni θ_i assumerà valori sullo spazio Θ della distribuzione G perché sono da essa campionati.

Consideriamo anche una partizione di Θ misurabile, per esempio $A_1,\dots,A_r$, e sia n_j il numero di valori osservati in A_j dove $n_j = \#\{i : \theta_i \in A_j\}$.

Riprendendo la distribuzione (2.4) e tenendo in mente il legame con la multinomiale, possiamo scrivere che:

$$(G(A_1),\dots,G(A_r))|\theta_1,\dots,\theta_n \sim Dir(\alpha H(A_1)+n_1,\dots,\alpha H(A_r)+n_r) \quad (2.13)$$

per tutte le partizioni finite e misurabili.

Se volessimo ricavare la distribuzione di base, bisogna procedere come fatto per la multinomiale e constatare che anche G è un processo di Dirichlet. A questo punto facendone il valor atteso si trova la distribuzione:

$$G(A)|\theta_1,\dots,\theta_n \sim Beta(\alpha H(A)+n_j,(\alpha+n)-(\alpha H(A)+n_j)) \quad (2.14)$$

dove il parametro di concentrazione è $\alpha+n$ e $n_j = \sum_{i=1}^{n} \delta_{\theta_i}(A_j)$ con δ_i punto di massa localizzato in θ_i. Quindi:

$$E(G(A)|\theta_1,\dots,\theta_n) = \frac{\alpha H + \sum_{i=1}^{n} \delta_{\theta_i}}{\alpha+n} \quad (2.15)$$

La distribuzione a posteriori è:

$$G|\theta_1,\dots,\theta_n \sim DP\left(\alpha+n, \frac{\alpha}{\alpha+n}H + \frac{n}{\alpha+n}\frac{\sum_{i=1}^{n}\delta_{\theta_i}}{n}\right) \quad (2.16)$$

Da cui è facile notare che, date le osservazioni, dal processo di Dirichlet otteniamo una famiglia di distribuzioni a priori coniugate e chiuse rispetto agli aggiornamenti a posteriori dei parametri. Inoltre, la posterior è costituita da una media pesata della distribuzione di base a priori H con peso $\frac{\alpha}{\alpha+n}$ e della distribuzione empirica $\frac{\sum_{i=1}^{n}\delta_i}{n}$ con peso $\frac{n}{\alpha+n}$. Quindi, all'aumentare di α, parametro di concentrazione, aumenta il peso dato alla a priori, mentre all'aumentare di n aumenta quello dato alla distribuzione empirica.

Possiamo fare un'osservazione: quando $\alpha \to 0$, la priori diventa non informativa. In questo caso la distribuzione predittiva di $\theta_1,\dots,\theta_n$ è data soltanto dalla distribuzione empirica. La distribuzione predittiva è la distribuzione di base a posteriori, questo aspetto verrà approfondito più avanti. Se $n \gg \alpha$, la distribuzione a posteriori ha un grosso peso attribuito alla distribuzione empirica e quest'ultima è una stretta approssimazione della vera distribuzione di fondo.

Vediamo ora, più nello specifico, le proprietà di questa distribuzione:

1. *Conjugacy.* Per dimostrare questa importante proprietà bisogna partire dalla distribuzione finito-dimensionale di una v.c. Dirichlet che è coniugata con la verosimiglianza multinomiale. Allora, la prior del processo di Dirichlet è anch'essa coniugata e permette di stimare una distribuzione non nota da dati che sono $i.i.d.$ Se $\theta_1,\dots,\theta_n$ sono $i.i.d.$ con distribuzione H, allora la distribuzione a posteriori di G

dati $\theta_1,\dots,\theta_n$ è un DP con misura centrale H e parametro di precisione $\alpha+\sum_{i=1}^{n}\delta_{\theta_i}$, dove ad α è stato sommato l'aggiornamento dovuto al campione estratto da $\theta_1,\dots,\theta_n$ tramite δ_{θ_i}.

Infatti, per ogni partizione misurabile finita $A_1,\dots,A_k$ e dati $\theta_1,\dots,\theta_k$ la distribuzione a posteriori è $(G(A_1),\dots,G(A_k))$ ha distribuzione Dirichlet k-dimensionale con parametri $\alpha H(A_j)+N_J$ con $N_j=\sum_{i=1}^{n}\mathbf{1}\{\theta_i\in A_j\}$ con $j=1,\dots,k$.

Considerando una partizione $\{B_1,\dots,B_m\}$ più fine di $\{A_1,\dots,A_k\}$ e la distribuzione a posteriori $(G(B_1),\dots,G(B_m))$ possiamo trovare marginalizzando la distribuzione $(G(A_1),\dots,G(A_k))$. Quindi, grazie alle proprietà della distribuzione di Dirichlet finito-dimensionale, abbiamo coincidenza nelle due posterior. Facendo partizioni sempre più fini e applicando il teorema in Appendice B.2, otteniamo che la proprietà è verificata. [20].

2. *Media a posteriori*. Se consideriamo quanto detto nella proprietà di coniugatezza, la distribuzione a posteriori combinata con la formula della media di un processo di Dirichlet porta ad avere una media:

$$E(G(A)|\theta_1,\dots,\theta_n)=\frac{\alpha}{\alpha+n}H+\frac{n}{\alpha+n}\frac{\sum_{i=1}^{n}\delta_{\theta_i}}{n} \tag{2.17}$$

   Notiamo che è una combinazione convessa che comprime la distribuzione empirica verso la prior. Per quanto riguarda i pesi che rientrano in questa media ponderata α e n, hanno un ruolo fondamentale e sono già stati discussi.

3. *Consistenza*. Giacché la distribuzione a posteriori si avvicina e converge alla vera distribuzione, ciò indica che la distribuzione è consistente.

   Più nel dettaglio, è una proprietà asintotica e possiamo definire:

   Definizione 3. *La distribuzione a posteriori è detta consistente se per un dato θ_0, o (θ_0,Π) è una coppia consistente dove Π è la prior, e per un vicinato (intorno) V di θ_0, $\Pi(\theta\notin V|data)\to 0$ (in probabilità o quasi certamente) come $n\to\infty$ quando θ_0 è il vero valore del parametro.*

   Innanzitutto, come si deduce dalla definizione, la consistenza è verificata quando abbiamo un campione molto grande che aiuta ad identificare correttamente il meccanismo generatore dei dati.

   Non si richiede una specifica prior affinché valga tale proprietà, piuttosto si fa riferimento alla famiglia cui essa appartiene.

   Per quanto riguarda questa proprietà, possiamo associare anche un tasso di convergenza, cioè accostare una misura che esprime a quale velocità avviene la convergenza al vero valore dove la probabilità a posteriori tende ad 1.

La proprietà di consistenza riporta al legame con la visione frequentista. Difatti, grazie a tale proprietà abbiamo l'esistenza di uno stimatore consistente. Finché essa è verificata per lo spazio convesso dei parametri, per lo spazio delle densità con metriche L_1, $Hellinger$ o altre metriche che richiedono un intorno convesso, allora la media a posteriori è uno stimatore consistente della vera media.

Esiste un teorema che assicura la consistenza per qualsiasi modello che abbia stimatori consistenti, il teorema di Doob (1948). Quando consideriamo un insieme nullo, non vi sono implicazioni circa il fatto che sia topologicamente piccolo. Può accadere che la prior sia degenere al punto θ^* e la consistenza non è verificata eccetto che per $\theta_0 = \theta^*$. Bisogna, a questo punto, saper mettere sufficienti condizioni sul vero valore e sulla prior affinché si abbia poi consistenza. Se il supporto del parametro è numerabile, il teorema di Doob implica che ci sia consistenza a qualunque punto a cui la prior assegni massa positiva.

Se stiamo trattando parametri su uno spazio finito-dimensionale la consistenza è quasi sempre garantita, i problemi sorgono con spazi infinito-dimensionali, dove seppur la prior assegna una probabilità positiva in un intorno, non è sufficiente a garantire la validità della proprietà.

Consideriamo il caso generico dove tutte le possibili coppie di veri parametri e prior portano alla consistenza, allora quando la dimensione è misurata topologicamente abbiamo una collezione molto ristretta. Un insieme F è chiamato $scarso$ ed è considerato essere topologicamente piccolo se può essere espresso come unione di insiemi C_i, $i \geq 1$, i cui complementari $\overline{C}_i$ sono vuoti [19]. L'esempio di Freedman (1963) [13] e la sua generalizzazione hanno dimostrato che le coppie che vanno bene sono scarse nel prodotto degli spazi. Ciò rappresenta un avvertimento sul corretto uso delle prior e sul dimostrare la validità dei teoremi di consistenza, non per questo però bisogna preoccuparsi in maniera spropositata visto che la maggior parte delle prior comunemente utilizzate incorpora le caratteristiche soggettive disponibili che permettono poi alla posterior di soddisfare le proprietà frequentiste per la consistenza.

La consistenza della posterior in un DP non è garantita in ogni applicazione. Se la posterior può essere espressa esplicitamente, allora è possibile non solo dimostrare la consistenza, ma anche darne il tasso di convergenza, cioè la velocità con cui converge, per mezzo della disuguaglianza di Chebyshev.

Se considerassimo la classe *tail-free* prior per cui è verificata la consistenza della posteriori e ci restringessimo allo spazio finito-dimensionale, la distribuzione a posteriori dipende solo dal conteggio delle zone (*celle*) corrispondenti a quelle in cui vale la proprietà di *tail-freeness*. In questo specifico caso, si riduce ad un problema di stima dei parametri di una distribuzione multinomiale. Un esempio di questo tipo di pro-

cesso è dato dal processo *Pólya tree*, che verrà trattato nel prossimo capitolo.

Vi è un altro approccio che è utile trattare. Esso si basa sulla teoria di Schwartz e, in particolare, sulla divergenza *Kullback-Leibler*.

Asserzione 1 (Kullback-Leibler divergence)**.** *La divergenza di Kullback-Leibler (anche detta entropia relativa) è una misura di come una distribuzione di probabilità è differente da una seconda distribuzione di probabilità di riferimento.*

Per stabilire la presenza di consistenza si può verificare se il numeratore dell'equazione (2.18) converge a 0 esponenzialmente come $e^{-\beta n}, \beta > 0$ e che il relativo denominatore converga a infinito per tutti i $\beta > 0$ in $e^{\beta n}$

$$\Pi(\theta \in B | X_1, \ldots, X_n) = \frac{\int_B \frac{p_{\theta,n}(X_1,\ldots,X_n)}{p_{\theta_0,n}(X_1,\ldots,X_n)} d\Pi(\theta)}{\int \frac{p_{\theta,n}(X_1,\ldots,X_n)}{p_{\theta_0,n}(X_1,\ldots,X_n)} d\Pi(\theta)} \tag{2.18}$$

dove $p_{\theta,n}(X_1,\ldots,X_n)$ è la distribuzione congiunta, $(X_1,\ldots,X_n)$ sono le variabili relative al campione, B è l'intorno, θ_0 è il vero valore del parametro θ e Π è la prior. Gli integrali fanno riferimento alla teoria di Schwartz, a riguardo si veda il capitolo 6 di Hjort(2010) [19].

Kullback-Leibler divergence può essere supportata dal *lemma di Fatou*.

Asserzione 2 (Fatou's lemma)**.** *Stabilisce una disuguaglianza che mette in relazione l'integrale di Lebesgue del limite inferiore di una sequenza di funzioni al limite inferiore degli integrali di queste funzioni.*

L'asserzione appena riportata impone una relazione chiave per la consistenza, viene indicata come condizione di positività della prior di Schwartz oppure proprietà di Kullback-Leibler della prior. Ciò vuol dire che la prior deve assegnare probabilità positiva a qualsiasi intorno del vero parametro e l'intorno è definito dalla vicinanza in termini di divergenza di Kullback-Leibler.

Per testare che ci sia consistenza uniformemente, Schwartz ideò un test che permettesse, controllando il numeratore dell'equazione (2.18), di testare se la probabilità di errore di I o II tipo andassero a 0 esponenzialmente. Si va a testare H_0, cioè $\theta = \theta_0$ impiegando la distanza di $Hellinger$, usata per quantificare la distanza tra due distribuzioni di probabilità similari. Negli spazi infinito-dimensionali però è difficile andare ad applicare il test direttamente.

Nonostante queste difficoltà, possono essere d'aiuto tecniche che troncano lo spazio dei parametri in base alla dimensione del campione.

Sebbene l'errore di II tipo non è molto problematico, non bisogna sottovalutare l'errore di I tipo. Un metodo è quello di considerare l'entropia, andando a verificare che la probabilità a posteriori si attenui ai valori limite della metrica, cioè ad un multiplo adeguatamente piccolo di n. Un'altra osservazione da fare è la seguente: una piccola probabilità a priori, non implica una piccola probabilità a posteriori. Tuttavia, se la prior probability è esponenzialmente piccola, allora lo è anche la posterior probability sotto condizione di positività di $Kullback-Leibler$. Questo segue da un'applicazione del teorema di Fubini (Appendice B.5).

4. *Limiti della posterior*. Consideriamo nuovamente l'equazione (2.16),

$$G|\theta_1,\ldots,\theta_n \sim DP\left(\alpha+n, \frac{\alpha}{\alpha+n}H + \frac{n}{\alpha+n}\frac{\sum_{i=1}^{n}\delta_{\theta_i}}{n}\right)$$

a questo punto teniamo n fisso e mandiamo $\alpha \to 0$. Il limite della posterior è chiamato *Bayesian bootstrap*. I campioni dal boostrap bayesiano sono distribuzioni discrete formati dai soli punti osservati i cui pesi sono distribuiti seguendo la distribuzione di Dirichlet. Allora, il *Bayesian bootstrap* può essere considerato come uno schema di ricampionamento più agevole del bootstrap di Efron [11].

Asserzione 3 (Efron's Bootstrap)**.** *Si basa sul fatto paradossale che l'unico campione disponibile serve per generarne molti altri e per costruire la distribuzione teorica di riferimento. Per poter conoscere la distribuzione si utilizza il principio plug-in (Principio di Sostituzione).*

Il secondo caso da prendere in considerazione è quando invertiamo il comportamento dei parametri, cioè α fisso e n variabile, abbiamo che il comportamento asintotico della media della posteriori dipende ed è controllato dalla distribuzione empirica.

Considerato che la distribuzione empirica converge uniformemente alla vera distribuzione, normalizzando abbiamo la convergenza al processo di ponte browniano.

Inoltre, tenendo presente le approssimazioni, sia A un qualsiasi insieme, la varianza di $G(A)$, per $n \to \infty$, diventa $O(n^{-1})$. La disuguaglianza di Chebyshev (Appendice B.6) implica che la distribuzione a posteriori di $G(A)$ si avvicina alla distribuzione degenere a $G_0(A)$, cioè la posterior di $G(A)$ è coerente a G_0, e la velocità di questa convergenza è $n^{-1/2}$ [19].

2.2.5 Distribuzione predittiva

Consideriamo nuovamente che $G \sim DP(\alpha, H)$ e che $\theta_1,\ldots,\theta_n$ sia un campione $i.i.d.$ estratto da G.

La distribuzione predittiva per θ_{n+1} può essere calcolata condizionando θ_{n+1} a $\theta_1,\ldots,\theta_n$ e marginalizzando rispetto a G.

Quando abbiamo parlato della distribuzioni a posteriori avevamo detto che la distribuzione di base a posteriori dati $\theta_1,\ldots,\theta_n$ era anche la distribuzione predittiva.

Infatti, possiamo scrivere:

$$P(\theta_{n+1} \in \mathbf{A}|\theta_1,\ldots,\theta_n) = E[G(\mathbf{A}|\theta_1,\ldots,\theta_n] = \\ = \frac{\alpha}{\alpha+n}H(\mathbf{A}) + \frac{1}{\alpha+n}\sum_{i=1}^{n}\delta_{\theta_i}(\mathbf{A}) \tag{2.19}$$

per ogni misurabile $\mathbf{A} \in \Theta$, date le prime n osservazioni. Come si può notare, l'equazione (2.19) dipende da G, per ottenere la distribuzione predittiva nel DP bisogna procedere integrando rispetto a G così da marginalizzare:

$$\theta_{n+1}|\theta_1,\ldots,\theta_n \sim \frac{\alpha}{\alpha+n}H + \frac{1}{\alpha+n}\sum_{i=1}^{n}\delta_{\theta_i} \tag{2.20}$$

Se volessimo interpretare la sequenza di distribuzioni predittive in termini di estrazioni da un'urna troveremmo il modello *Pólya urn*, trattato nei successivi capitoli.

La predittiva può essere molto utile per campionare e fare inferenza in un processo di Dirichlet.

3
Modelli di Mistura

Presenteremo alcuni modelli che utilizzano il processo di Dirichlet come base. Questi processi sono utili come prior in diverse situazioni.

In questo capitolo verranno presentate le misture di DP, verrà spiegata la rappresentazione stick breaking e, infine, una loro estensione (estensione gerarchica).

Tratteremo anche le conseguenze di avere delle osservazioni che non siano $i.i.d$.

3.1
Mixture models

Un'altra proprietà di cui non si è parlato nel precedente capitolo è la proprietà di *clustering*. Infatti, l'analisi dei cluster è un problema ricorrente in statistica, cioè si vogliono trovare eventuali pattern e raggruppare unità statistiche mutualmente simili tra loro e dissimili da quelle in altri gruppi.

Non solo esistono algoritmi di partizione per soddisfare tale scopo (come le *K-medie*), metodi gerarchici (con legame singolo, completo o criterio di Ward con unità miste), ma anche modelli di mistura basati su metodi probabilistici.

In un problema di clustering, quindi, dati $x_1,\dots,x_n$ si ha l'obbiettivo di suddividere il campione in gruppi, ma si richiede l'utilizzo del metodo ideale per suddividere le unità in subset in base alle variabili che si hanno nel dataset e per trovare il numero adeguato di gruppi presenti nei nostri dati.

Sia $\mathbf{x} = (x_1,\dots,x_n)$ l'insieme di dati, l'assunzione basica di un clustering è che ogni x_i appartiene ad un singolo cluster. Con i *modelli di mistura* si vuole trovare come è stato generato l'insieme di dati. Difatti, si può pensare che ogni gruppo sia composto da oggetti che appartengono alla stessa distribuzione, ma che si differenziano per i parametri fondamentali di ogni distribuzione (e.g. media e varianza per la distribuzione Normale). Inoltre, essi hanno una loro probabilità di essere rappresentati.

Definizione 4 (Modelli di Mistura)**.** *I modelli di mistura sono un tipo di modello di densità costituito da un certo numero di funzioni di densità, se per esempio fossero densità gaussiane si chiamerebbe* Gaussian Mixture Models*, e queste funzioni sono unite per fornire una densità multimodale. Il* mixture model *serve, quindi, per modellare una distribuzione di probabilità come somma di distribuzioni.*

Se il numero di cluster è finito, pertanto c'è un numero finito K di probabilità non-nulle, la mistura viene chiamata **mistura finita**.

Possiamo scrivere l'assegnazione ad ogni cluster tramite una variabile casuale z_i, dove $z_i = k$ indica che x_i appartiene al cluster k. Finché non si

hanno le assegnazioni ai cluster, queste variabili rimangono non osservate. A livello empirico è difficile specificare un'esatta distribuzione, allora specifichiamo in modo più naturale una famiglia parametrica con parametri ignoti che rappresenta il meccanismo con cui si generano i dati. Questi parametri non noti della componente di mistura vengono indicati con θ_{z_i}.

Le misure centrali possono, quindi, contenere iperparametri che vanno a supporto della stima dei veri parametri della distribuzione condizionandosi ad essi e poi integrando per eliminarli dall'analisi.

Le misture di processi di Dirichlet vengo anche abbreviate con MDP (*Mixture of Dirichlet Process models*).

Le MDP mantengono alcune delle proprietà del DP, come il fatto che i campioni siano discreti, ma le proprietà di *self-similarity* e di *tail-freeness* non vengono rispettate.

Nel caso in cui la prior fosse MDP, anche la posterior lo sarebbe. Ciò accade perché condizionandosi agli iperparametri, rimane preservata la proprietà di coniugatezza. Questo può essere dimostrato facilmente se ci riferiamo a $\Pi(\theta)$ per la prior, a $\Pi(\theta|data)$ per la posterior e a g_θ per la distribuzione di densità dei dati generati da G. Grazie al teorema di Bayes per il caso parametrico con variabili casuali continue, possiamo scrivere:

$$\Pi(\theta|data) \propto \Pi(\theta) \prod_{i=1}^{n} g_\theta(x_i) \tag{3.1}$$

E' una conseguenza dello schema dell'urna di Blackwell-MacQueen che descrive la distribuzione congiunta assumendo tutte le X_i distinte.

Andando a trattare un MDP più nello specifico, le componenti di questo modello sono un numero infinito numerabile. Consideriamo $\mathbf{x}$ vettore di osservazioni, siano $\{\theta_1,\ldots,\theta_n\}$ l'insieme di parametri latenti. Quest'ultimi hanno una distribuzione a priori $G \sim DP(\alpha,H)$. Ogni θ_i è, quindi, un'estrazione da questo processo ed è indipendente e con stessa forma rispetto alle altre estrazioni. Invece, ogni osservazione x_i ha distribuzione $F(\theta_i)$ parametrizzata da θ_i.

Possiamo quindi definire le seguenti distribuzioni condizionate:

$$\begin{aligned} x_i|\theta_i &\sim F(\theta_i) \\ \theta_i|G &\sim G \\ G|\alpha,H &\sim DP(\alpha,H) \end{aligned} \tag{3.2}$$

Una considerazione che possiamo fare deriva dal fatto che G sia discreta, infatti, i multipli di θ_i possono assumere lo stesso valore simultaneamente. Allora, il modello precedentemente scritto può essere visto come un modello di mistura dove le x_i che hanno lo stesso valore di θ_i apparterranno allo stesso cluster.

Possiamo chiamare π il vettore delle proporzioni di mistura, cioè $\pi = \{\pi_{z_i}\}_{z_i=1}^{K}$, dove, come detto precedentemente, le z_i sono le variabili latenti e l'osservazione i-esima appartiene al gruppo k quando $z_i = k$. In questo modo possiamo descrivere il processo generatore:

1. selezionare uno dei K gruppi con probabilità π_{z_i};
2. generare x_i dalla componente di mistura corrispondente parametrizzata da θ_{z_i}.

Pertanto, dati i parametri della distribuzione, si può definire la probabilità del dataset:

$$p(\mathbf{x}|\theta) = \prod_{i=1}^{n} \sum_{z_i=1}^{K} \pi_{z_i} p(x_i|z_i, \theta_{z_i}) \tag{3.3}$$

dove la densità di probabilità della componente associata al gruppo z_i è indicata da $\pi_{z_i} p(x_i|z_i, \theta_{z_i})$. Dato che in questo caso stiamo trattando i nostri dati con un modello di tipo parametrico, si procede con l'usuale stima di massima verosimiglianza al fine di stimare i parametri. Siccome si tratta di un modello di mistura bisogna, innanzitutto, calcolare la probabilità condizionata ai valori dei parametri e verificare che ogni osservazione appartenga ad un certo gruppo, utilizzando le stime iniziali dei parametri:

$$P(z_i = k|x_i) = \frac{\pi_k P(x_i|\theta_k)}{\sum_l \pi_l P(x_i|\theta_l)} \tag{3.4}$$

Poi si procede ad aggiornare le stime tramite le probabilità precedentemente calcolate.

Una metodologia come questa utilizzata per l'individuazione dei cluster è molto più flessibile. Con l'aggiornamento dei parametri di media e varianza, ad esempio, giungiamo a stimare la forma dei cluster poiché ne deduciamo il modo in cui i punti si distribuiscono per mezzo della varianza e il punto attorno al quale lo fanno tramite la media.

Un contro di questo approccio rimane che con l'aumento della flessibilità del modello si possono avere problemi di overfitting e quindi non possiamo generalizzare i risultati trovati. Per evitare che ciò accada si può ricorrere a stimare la probabilità a posteriori sfruttando la regola di Bayes:

$$p(\theta|\mathbf{x}) = \frac{p(\theta)p(\mathbf{x}|\theta)}{\int p(\theta)p(\mathbf{x}|\theta)d\theta} = \frac{p(\theta)p(\mathbf{x}|\theta)}{p(\mathbf{x})} \tag{3.5}$$

Con questa scrittura si può, poi, sostituire la funzione di verosimiglianza e arrivare a stimare i parametri con metodi di stima di tipo MCMC. Questi metodi possono essere utili per simulare le variabili latenti dalla posterior.

I kernel che vengono utilizzati in un MDP possono essere scelti in base allo scopo. Ad esempio, se la funzione da stimare è lineare, si può utilizzare un location-scale kernel come la densità normale. Se invece è semi-lineare allora le distribuzioni appropriate sono *Weibul, Gamma e Log-Normale*. Se, invece, si ha un'unità intervallare allora dovremo utilizzare una distribuzione *Beta*. Proseguendo fino a modelli più complicati.

3.1.1 Numero di componenti in una mistura

Asserzione 4. *Le misture nonparametriche bayesiane sono anche uno strumento per selezionare automaticamente il numero di componenti in una mistura.*

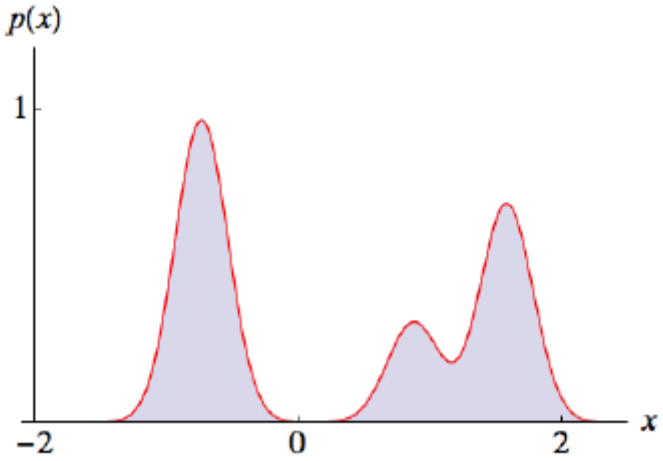


Figura 3.1: Una mistura gaussiana con misura di mistura θ. Ogni densità parametrica $p(x_i|\phi)$ è $Normale$ con varianza = 0.2 e media ϕ [27].

Questo vuol dire che se si ha una MDP su un campione di ampiezza n, per ogni soluzione di cluster supportata dalla posterior, troviamo un numero random, finito di cluster $K \leq n$ [27]. Trovare una posterior su un numero di cluster non è propriamente selezione del modello. Nel DP si hanno assunzioni implicite sul modello:

Asserzione 5. *Per $n \to \infty$, inevitabilmente (con probabilità 1) osserviamo un numero infinito di cluster.*

Quindi, per integrare quanto detto precedentemente, al fine di scegliere la strategia adeguata per il numero di componenti bisogna distinguere tre problemi:

(1) K è *finito e noto*: assunzione data da un modello di misture finito di ordine K.

(2) K è *finito e non noto*: mistura finita di ordine non noto. Questo problema non viene risolto tramite un DP o un'altra mistura finita.

(3) K è *infinito*: assunzione fatta da un DP.

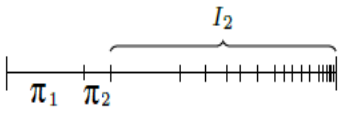
Se generiamo un campione da una mistura finita, e successivamente ne calcoliamo la posterior sotto un DP infinito, allora asintoticamente la posterior si concentrerà su un numero infinito di cluster. Un campione infinitamente grande avrà un numero infinito di cluster quasi certamente.

3.2 Stick-breaking

Quando si ha un numero infinito di componenti di mistura, se anche avessimo variabili $i.i.d.$ e le normalizzassimo, non potremmo ottenere dei risultati adeguati con la procedura descritta precedentemente, poiché una somma

infinita di variabili indipendenti e identicamente distribuite diverge. Però si può pensare ad una soluzione molto semplice, cioè possiamo campionare π_1 da una distribuzione di probabilità distribuita su $[0,1]$. Una volta osservato π_1 possiamo campionare π_2 che prenderà valori su $[0,1-\pi_1]$ e si può proseguire così fino ad esempio a π_k.

Quindi, ad esempio, per $k=2$ si ha



dove $I_2 := [0,1-(\pi_1+\pi_2)]$.

La costruzione di π può essere pensata come se prendessimo il bastoncino di lunghezza unitaria e lo spezzassimo ad una lunghezza pari a π_1 in corrispondenza del punto β_1. Poi spezziamo ricorsivamente la porzione rimanente prima a lunghezza π_2, poi π_3, ..., ottenendo come punti di rottura $\beta_2, \beta_3, \ldots$ e così via.

Possiamo formalizzare il tutto, ricordandoci che le estrazioni da $G \sim Dir(\alpha, H)$ sono discrete e con probabilità uno. Questa distribuzione può essere vista come una somma pesata di punti di massa poiché le estrazioni da G possono assumere lo stesso valore con probabilità positiva. Quindi, è una distribuzione discreta e per averne una rappresentazione migliore possiamo utilizzare la cosiddetta *stick-breaking representation* introdotta da *Sathuraman* (1994) [37].

Date delle sequenze indipendenti di variabili $i.i.d.$ $(\beta_k)_{k=1}^{\infty}$ e $(\theta_k^*)_{k=1}^{\infty}$:

$$\begin{aligned} &\beta_k \sim Beta(1,\alpha) && \theta_k \sim H \\ &\pi_k = \beta_k \prod_{l=1}^{k-1}(1-\beta_l) && G = \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k^*} \end{aligned} \tag{3.6}$$

Per quanto detto finora possiamo scrivere $\pi \sim GEM(\alpha)$ dove le lettere stanno per *Griffiths, Engen e McCloskey* [28].

Grazie a questa rappresentazione è possibile avere delle estensioni nell'applicabilità del processo di Dirichlet. Difatti, *Sethuraman* ha permesso di generare un'approssimazione del processo di Dirichlet attraverso un'appropriato troncamento ad uno stadio finito. Questo è utilissimo quando analiticamente non è possibile derivare le espressioni e la posterior può essere calcolata solo tramite simulazioni. Quando imponiamo questo troncamento, si può trattare il problema come nel caso parametrico e, quindi, sfruttare tecniche MCMC. Inoltre, un altro vantaggio che segue da questa rappresentazione è che si possono costruire nuove misure random cambiando la distribuzione dello *stick-breaking* da una $Beta(1,\alpha)$ ad altre distribuzioni. Ad esempio, *two-parameter Poisson-Dirichlet process* dove la distribuzione *stick-breaking* varia con lo stadio considerato.

Un altro beneficio di questa rappresentazione, per applicazioni con covariate, è la possibilità dell'introduzione di dipendenza attraverso diver-

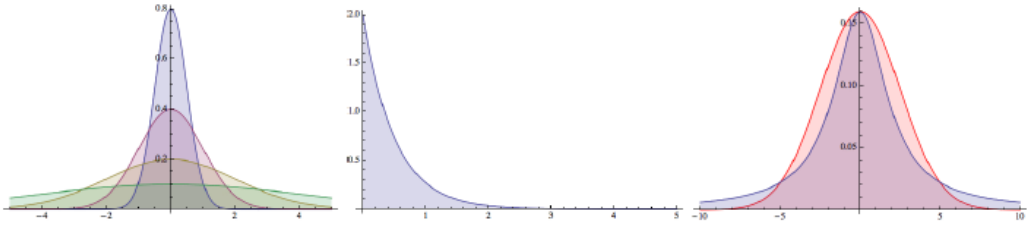


Figura 3.2: Esempi di misture [27].

se misure randomiche che marginalmente sono DP permettendo che ci sia dipendenza all'interno del supporto, o per i loro pesi o per entrambi [19].

3.3
Modelli di mistura con costruzione stick-breaking

Sia z_i la variabile di assegnamento al gruppo. Ricordiamo che assume valore k con probabilità π_k. G è discreta, i diversi θ_i possono assumere gli stessi valori, in questo caso le osservazioni x_i con lo stesso θ_i provengono dalla stessa componente di mistura e appartengono allo stesso cluster.

$$\begin{aligned} \boldsymbol{\pi}|\alpha &\sim GEM(\alpha) & \theta_k^*|H &\sim H \\ z_i|\boldsymbol{\pi} &\sim Mult(\pi) & x_i|z_i,\{\theta_k^*\} &\sim F(\theta_{z_i}^*) \end{aligned} \tag{3.7}$$

dove $\theta_i = \theta_{z_i}^*$ e $G = \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k^*}$. Nella terminologia dei modelli di mistura, si ha che i θ_k^* sono i parametri dei gruppi, $\boldsymbol{\pi}$ rappresenta il vettore delle proporzioni di mistura, $F(\theta_k^*)$ è la distribuzione dei dati del gruppo k e H è la distribuzione a priori sui parametri dei gruppi. Nella Figura 3.3 è rappresentato il modello con le dipendenze fra variabili che è possibile trovare in un modello di mistura. Il modello basato sul processo di Dirichlet può essere visto come un modello mistura con un numero di gruppi numerabile e illimitato. Può essere mossa un'osservazione riguardante il numero di cluster: siccome i π_k decrescono rapidamente in maniera esponenziale, solo un piccolo numero di cluster verrà utilizzato per i dati a priori. Il numero atteso di componenti a priori sono proporzionali logaritmicamente al numero di osservazioni $O(\alpha \log(n))$.

In un modello di mistura DP il numero di cluster non è fissato, viene automaticamente scelto dei dati attraverso l'inferenza bayesiana sulla posterior. Quindi, questi modelli sono un'alternativa ai modelli di mistura finiti che possono avere diverse difficoltà.

Partendo da una sequenza di modelli di mistura finiti, si può ricavare il modello basato sul DP come *limite* della sequenza, andando a portare ad infinito il numero delle componenti di mistura.

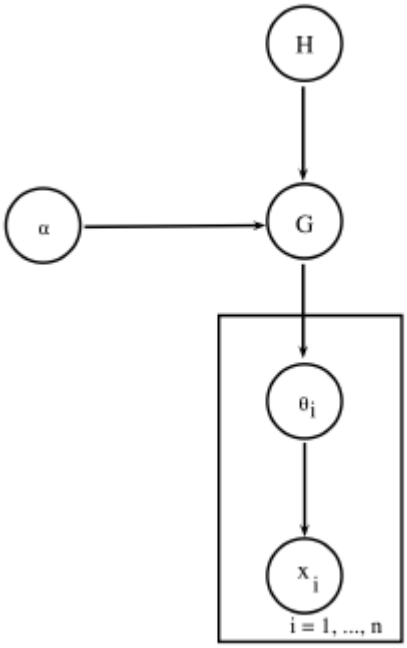


Figura 3.3: Dipendenza tra variabili in un modello di mistura.

3.4 Osservazioni non i.i.d.

Il fatto di aver richiesto osservazioni $i.i.d.$ porta, ovviamente, ad avere una facilitazione a livello computazionale, come anche richiederle il più vicino possibili a questa situazione. Questo è importante perché la dipendenza nella prior rende difficile riuscire ad ottenere la posterior sia computazionalmente (in termini matematici e temporali) sia statistici (per l'ammontare di dati richiesti). Nel caso in cui avessimo osservazioni non indipendenti, non si può renderle indipendenti automaticamente, ma bisogna scegliere una forma specifica di dipendenza che introduca un'assunzione sul modello da noi costruito.

Nonostante questo, è possibile estendere la proprietà di consistenza della posteriori anche in caso di non identica distribuzione o di dipendenza. Ciò avviene allo stesso modo con cui si può applicare la legge debole dei grandi numeri alle somme dei rapporti di log-verosimiglianza, e perché se viene a mancare solo l'identica distribuzione è possibile applicare la legge forte di Kolmogorov. Inoltre, sono presenti dei test che permettono di capire il modo appropriato con cui gestire questi problemi.

Questa estensione è molto utile nelle stime di regressioni non parametriche con covariate fisse, stime nello spettro delle densità e nelle serie storiche usando la verosimiglianza *Whittle*, molto utile poiché approssima la verosimiglianza nelle serie storiche gaussiane stazionarie.

3.5 Modelli gerarchici

Dopo aver illustrato il processo di Dirichlet, le sue proprietà e alcuni modelli che ne derivano, in questo paragrafo ci occuperemo di dare un'interpretazione più approfondita dei modelli di mistura derivandone un'estensione gerarchica.

Grazie all'approccio bayesiano, infatti, possiamo estendere i modelli di mistura in modelli gerarchici, questo vale anche per MDP. L'estensione a questi tipi di modelli consente di estendere il processo di Dirichlet in "livelli di gerarchie" ed utilizzarlo anche per problemi di classificazione dove vogliamo creare cluster che nei diversi insiemi siano condivisi.

Il principio di questa estensione consiste nel considerare un processo di Dirichlet avente come distribuzione di base un altro processo di Dirichlet.

$$\begin{aligned} G &\sim DP(\alpha, G_0) \\ G_0 &\sim DP(\gamma, H) \end{aligned} \tag{3.8}$$

La scrittura soprastante descrive come la costruzione ricorsiva di G implica che il suo supporto sia discreto e determinato da G_0. Quest'ultima viene definita *Hierarchical Dirichlet Process* (HDP) [38].

Consideriamo J gruppi di osservazioni e sia θ_{ji} il parametro latente associato all'osservazione i-esima del gruppo j-esimo, x_{ji}. Il processo definisce un insieme di misure di probabilità G_j, una per ogni gruppo, ovvero stiamo definendo un processo di Dirichlet G_j per ogni insieme j, e una misura globale casuale di probabilità G_0. Quindi, questa collezione di processi $\{G_j\}$ è definita su uno spazio di probabilità comune. La misura globale G_0 è distribuita come un processo di Dirichlet e rappresenta la distribuzione di base di questa collezione. Quindi, grazie al processo gerarchico si ha un legame dal punto di vista probabilistico tra tutti i G_j.

Le distribuzioni che ne derivano sono:

$$\begin{aligned} G_0|\gamma, H &\sim DP(\gamma, H) \\ G_j|\alpha, G_0 &\sim DP(\alpha, G_0) \qquad j = 1, \ldots, J \end{aligned} \tag{3.9}$$

dove per G_0 si ha che γ è il parametro di concentrazione e H la misura base di probabilità, mentre per G_j gli stessi parametri sono α e G_0. Questo vuol dire che G_0 è il processo base di ogni G_j e la sua distribuzione di base H è la prior dei parametri θ_{ji}. Siccome ogni G_j eredita l'insieme di atomi dal processo "padre" G_0, allora è possibile la condivisione dei punti di massa. Se si suppone che ci sia diversa variabilità tra i gruppi, possiamo usare un diverso parametro di concentrazione α_j per ogni gruppo j. In questo caso, aggiungendo la verosimiglianza, si ha che il modello completo risulta essere:

$$\begin{aligned} G_0|\gamma, H &\sim DP(\gamma, H) \\ G_j|\alpha, G_0 &\sim DP(\alpha, G_0) \\ \theta_{ji}|G_j &\sim G_j \\ x_{ji}|\theta_{ji} &\sim F(\theta_{ji}) \end{aligned} \tag{3.10}$$

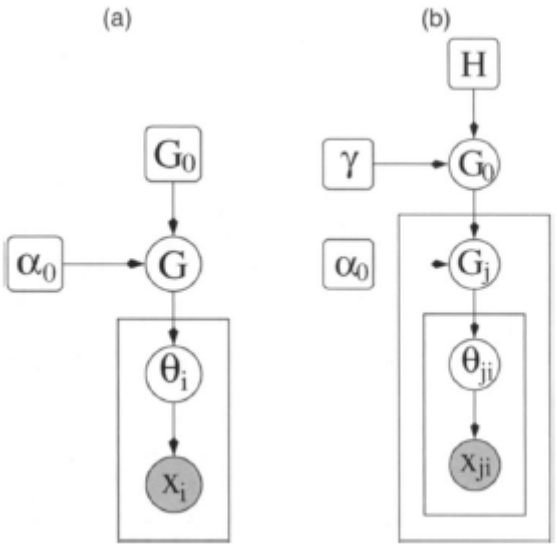


Figura 3.4: Rappresentazione grafica di un modello mistura di un processo di Dirichlet (a) e di un modello di mistura gerarchico di un processo di Dirichlet (b). Ogni nodo nel grafico è associato ad una variabile casuale, dove i nodi a sfondo scuro denotano le variabili osservate. I rettangoli, invece, denotano replicazioni del modello all'interno del rettangolo stesso. A volte il numero delle replicazioni viene dato nell'angolo in basso a destra del rettangolo [38].

Questa è la definizione completa di *Hierarchical DP Mixture model*. Possiamo vedere anche graficamente un confronto rispetto ai semplici modelli di mistura nella Figura 3.4.

L'HDP può essere esteso a più di due livelli poiché possiamo avere che la misura di base H è anch'essa un DP. E' possibile estendere la gerarchia a tanti livelli quanti ne siano ritenuti utili. In generale, otteniamo un albero dove ad ogni nodo è associato un processo di Dirichlet e i "figli" di un nodo sono condizionatamente indipendenti dai genitori che svolgono da misura base per essi.

Siccome G_0 è distribuito come un processo di Dirichlet possiamo rappresentare il modello per mezzo della costruzione *stick-breaking*. Siano

$$\begin{aligned} G_0 &= \sum_{k=1}^{\infty} \beta_k \delta_{\theta_k^*} \\ \theta_k^* | H &\sim H \end{aligned} \tag{3.11}$$

dove $\boldsymbol{\beta} = (\beta_k)_{k=1}^{\infty} \sim GEM(\gamma)$ sono mutualmente indipendenti. Dal momento che G_0 ha supporto sui punti $\boldsymbol{\theta} = (\theta_k)_{k=1}^{\infty}$ e G_j avrà lo stesso supporto, allora possiamo rappresentarlo nel seguente modo:

$$G = \sum_{k=1}^{\infty} \pi_{jk} \delta_{\theta_k^*} \tag{3.12}$$

Siano dati i pesi $\boldsymbol{\pi}_j = (\pi_{jk})_{k=1}^{\infty}$. Essi risultano indipendenti dato $\boldsymbol{\beta}$ per il fatto che i G_j sono indipendenti dato G_0. Attraverso alcuni semplici passaggi possiamo verificare che anche i pesi si distribuiscono come un processo di Dirichlet.

Siano $(A_1,\dots,A_r)$ una partizione di Θ e $K_l = \{k : \theta_k \in A_l\}$ per $l = 1,\dots,r$. Possiamo notare che $(K_1,\dots,K_r)$ è una partizione finita di interi positivi. Per ogni j, abbiamo:

$$\begin{aligned} &(G_j(A_1)),\dots,G_j(A_r)) \sim Dir(\alpha G_0(A_1),\dots,\alpha G_0(A_r)) \\ \Rightarrow &\left(\sum_{k\in K_1}\pi_{jk},\dots,\sum_{k\in K_r}\pi_{jk}\right) \sim Dir\left(\alpha\sum_{k\in K_1}\beta_k,\dots,\alpha\sum_{k\in K_r}\beta_k\right) \end{aligned} \tag{3.13}$$

per ogni partizione finita di interi positivi [38]. Allora, $\boldsymbol{\pi}_j$ si distribuisce come un processo di Dirichlet $DP(\alpha,\boldsymbol{\beta})$, dove $\boldsymbol{\beta}$ e $\boldsymbol{\pi}_j$ possono essere interpretati come misure di probabilità su interi positivi.

Possiamo scrivere il modello della (3.10) con la rappresentazione *stick-breaking*:

$$\begin{aligned} \boldsymbol{\beta}|\gamma &\sim GEM(\gamma) \\ \boldsymbol{\pi}_j|\alpha,\boldsymbol{\beta} &\sim DP(\alpha,\boldsymbol{\beta}) \\ z_{ji}|\boldsymbol{\pi}_j &\sim \boldsymbol{\pi}_j \\ \theta_k^*|H &\sim H \\ x_{ji}|z_{ji},\{\theta_k^*\} &\sim F(\theta_{z_{ji}}) \end{aligned} \tag{3.14}$$

dove θ_k^* è il parametro che caratterizza la componente di mistura k e, quindi, il cluster, invece z_{ji} è la variabile di assegnamento al gruppo, associata all'osservazione x_{ji}. Quando le osservazioni x_{ji} sono associate a valori uguali del parametro θ_{ji}, allora esse saranno generate dalla stessa distribuzione, anche se appartengono a gruppi differenti.

Gli atomi ad ogni nodo nella rappresentazione *stick-breaking* sono così condivisi tra tutti i nodi discendenti, introducendo la nozione di cluster condivisi a molteplici livelli di risoluzione.

Per intuire al meglio come funziona la condivisione tra cluster, si rimanda al capitolo successivo. In particolare ai paragrafi relativi all'*Indian buffet* e al *Chinese Restaurant Franchise*.

4
Oltre il Processo di Dirichlet

Presenteremo ora un excursus di alcuni processi che vanno oltre il processo di Dirichlet. I principali, notevoli di menzione, sono delle predittive come *Pólya urn* (già citata quando parlavamo della costruzioni di campioni), il *ristorante cinese*, il *buffet indiano*, il *franchise di ristoranti cinesi*, nuove prior come *Pólya trees*, e il *processo gaussiano* che combinato con il processo di Dirichlet permette nuovi tipi di applicazione dei modelli bayesiani con un approfondimento di due casi separati di *supersmooth* e *ordinary smooth*. Tutti questi processi hanno come base il processo di Dirichlet e si sviluppano in direzioni differenti.

4.1
Pólya urn

Quando siamo interessati al campionamento da G, possiamo sfruttare la stessa metodologia utilizzata per *Pólya urn*.

Per parlare di questo processo bisogna pensare alla sequenza di distribuzioni predittive (2.20)

$$\theta_{n+1}|\theta_1,\dots,\theta_n \sim \frac{\alpha}{\alpha+n}H + \frac{1}{\alpha+n}\sum_{i=1}^{n}\delta_{\theta_i}$$

per $\theta_1,\theta_2,\dots$ che può essere interpretata come un *Pólya urn model*, cioè un modello di urna semplice. Consideriamo un'urna contenente palline di diverso colore. Ogni valore di Θ rappresenta differenti colori, mentre le palline sono rappresentate dalle estrazioni θ da G i cui valori sono, appunto, i colori.

All'inizio l'urna è vuota e sarà riempita man a mano con le estrazioni fatte fino a quel momento. Quindi estraiamo una prima pallina, $\theta_1 \sim H$ che verrà colorata con un certo colore e sarà inserita nell'urna. Questo verrà ripetuto nei passi successivi. Al passo $n+1$-mo con probabilità $\frac{\alpha}{\alpha+n}$ si può estrarre un nuovo colore, quindi $\theta_{n+1} \sim H$, e dipingere la pallina con quel colore e metterla nell'urla, oppure con probabilità $\frac{n}{\alpha+n}$ si può estrarre dall'urna una pallina, quindi θ_{n+1} viene estratta dalla distribuzione empirica, e dipingere la nuova pallina dello stesso colore di quella estratta e inserirle entrambe nell'urna.

Lo schema dell'urna è molto utile in quanto genera sequenze scambiabili.

Si può generalizzare questo modello con una semplice immagine (4.1).

I punti salienti dell'algoritmo sono:

1. Inizialmente l'urna è vuota.
2. Colora una pallina e sostituiscila due volte nell'urna.
3. La pallina determina lo stato successivo dell'urna.

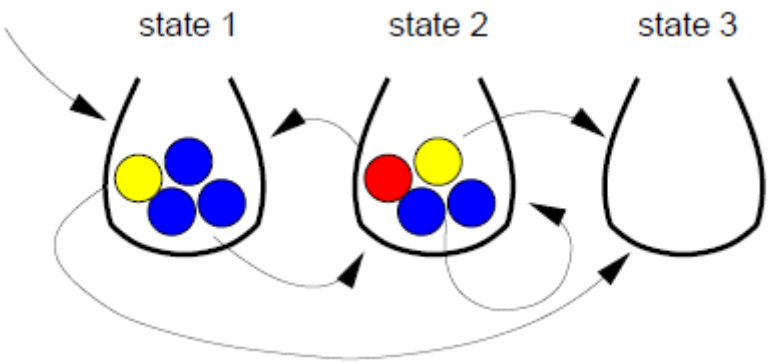


Figura 4.1: Generalizzazione dello schema dell'urna.

4. Se lo stato successivo non è ancora stato riempito con una pallina, torna al punto 2.

Questo semplice schema riassuntivo, illustra come i vari stati tra loro si influenzino, quindi conferma il motto: *"Stay more likely on the beaten path"* ("Rimani maggiormente sul sentiero già battuto"), poiché gli stati visitati maggiormente sono quelli con probabilità più alta di essere visitati nuovamente. Questo modo di muoversi tra gli stati viene anche detto *"Bernoulli trips on multiple states"*.

4.2 Pólya trees

Il processo *Pólya tree* è già stato menzionato in precedenza nella trattazione della proprietà di *tail-freeness*.

Per stimare la densità utilizzando tale processo bisogna per semplicità considerare delle partizioni binarie. Esse vengono utilizzate nella distribuzione di massa nella struttura dell'albero ottenuto sequenzialmente dalla mediana, dai quartili, dai percentili, … di una densità. Possiamo assumere che i parametri della distribuzione $Beta$ utilizzata per gli split siano gli stessi quando si considera lo stesso livello dell'albero. Quindi, si considerano le seguenti distribuzioni:

$$\alpha_{\epsilon 0} \sim Beta(\beta_{\epsilon 0}, \beta_{\epsilon 1}) \qquad \alpha_{\epsilon 1} = 1 - \alpha_{\epsilon 0} \tag{4.1}$$

Il parametro ϵ può valere ad esempio 0.01, 0.05, …

Definizione 5 (Pólya tree)**.** *Una distribuzione P è distribuita secondo un* Pólya tree *(PT)*

$$P \sim PT(B_\epsilon, \beta_\epsilon) \tag{4.2}$$

se e solo se B_ϵ è una partizione gerarchica di una linea reale e per ogni ϵ esiste una variabile casuale

$$\alpha_{\epsilon 0} \sim Beta(\beta_{\epsilon 0}, \beta_{\epsilon 1}) \tag{4.3}$$

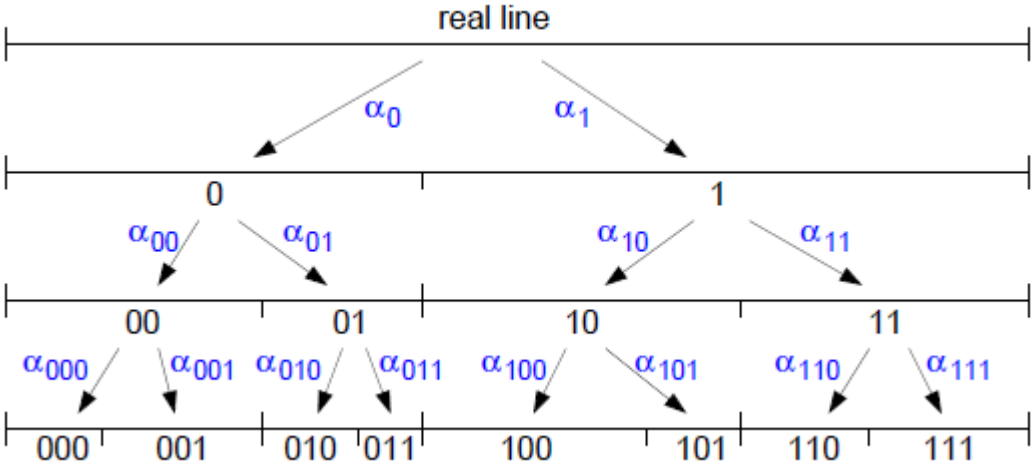


Figura 4.2: Rappresentazione di un *Pólya tree*.

e per ogni $m = 1,2\ldots$ *e per ogni* $\epsilon = \epsilon_1\epsilon_2\ldots\epsilon_m$ *si ha che:*

$$P(B_\epsilon) = \left(\prod_{j=1}^{m} \delta_0(\epsilon_j)\alpha_{\epsilon_1\epsilon_2\ldots\epsilon_{j-1}0}\right)\left(\prod_{j=1}^{m} \delta_1(\epsilon_j)(1-\alpha_{\epsilon_1\epsilon_2\ldots\epsilon_{j-1}1})\right) \tag{4.4}$$

Se il parametro β venisse scelto tale che $\beta_{\epsilon 0} = \beta_{\epsilon 1}$ la magnitudine del parametro β_ϵ controlla la variabilità. Possiamo quindi ottenere il seguente risultato:

$$E(P) = P_0 \quad \text{per} \quad P \sim PT(B_\epsilon, \beta_\epsilon) \tag{4.5}$$

Il teorema di Kraft (1964) assicura che una distribuzione generata da un *Pólya tree* ammette densità quasi certamente se la somma dell'inverso dei parametri della distribuzione *Beta* ad ogni livello dell'albero convergono. Per garantire che valga la proprietà di Kullback-Liebler per la consistenza della posterior, bisogna rendere più restrittiva la condizione di convergenza richiedendo la convergenza della somma dell'inverso dei parametri sotto radice quadrata.

Uno svantaggio di questa metodologia è che risulta difficile controllare lo spazio dove la prior è essenzialmente supportata a causa della mancanza di regolarità. Consente, però, di mantenere la proprietà di coniugatezza.

4.3 Chinese restaurant process

Per introdurre questo processo dal nome particolare, dobbiamo considerare quanto detto precedentemente sui cluster. Con riferimento alla formula (2.20) avevamo constato che un'estrazione da G può avere una probabilità positiva di assumere un valore uguale ad una delle precedenti. Bisogna però sottolineare che c'è un effetto di rafforzamento positivo, cioè se un valore è stato estratto spesso, è molto più probabile che venga estratto nuovamente. Questo deriva dalla proprietà di clustering del DP.

I valori unici di $\theta_1,\dots,\theta_n$ inducono un partizionamento del set $[n] = \{1,\dots,n\}$ nei cluster. Chiamiamo quindi $\theta_1^*,\dots,\theta_m^*$ i valori unici tra $\theta_1,\dots,\theta_n$ e allora per ogni θ_k^* sia n_k il numero di ripetizioni. A questo punto possiamo riscrivere la distribuzione predittiva (2.20) tenendo conto dei valori unici e delle loro ripetizioni, nel seguente modo:

$$\theta_{n+1}|\theta_1,\dots,\theta_n \sim \frac{\alpha}{\alpha+n}H + \frac{1}{\alpha+n}\sum_{k=1}^{m} n_k \delta_{\theta_k^*} \tag{4.6}$$

Dalla formula appena scritta emerge che maggiore è n_k, maggiore sarà la probabilità che esso cresca perché i valori θ_k^* rientrano con peso proporzionale a n_k in θ_{n+1}.

Questa è la classica situazione dove "i ricchi si arricchiscono", quindi i cluster grandi crescono con velocità maggiore rispetto ai cluster più piccoli.

Dato che la sequenza $\theta_1,\dots,\theta_n$ è una sequenza casuale, anche la partizione di $[n]$ sarà casuale. Il partizionamento mantiene tutte le proprietà del DP.

Con *Chinese restaurant process* (CRP) viene comunemente indicata la distribuzione delle partizioni. In questa metafora, in un ristorante cinese ci sono un numero infinito di tavoli a cui possono sedersi un numero infinito di clienti.

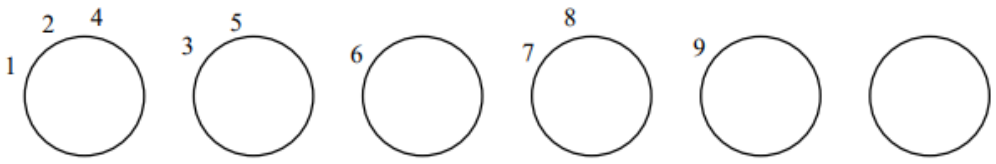


Figura 4.3: Esempio di tavoli al ristorante cinese, dove i numeri rappresentano clienti differenti.

Inizialmente tutti i tavoli sono vuoti. Il primo cliente entra nel ristorante e si siede al primo tavolo. Il secondo cliente entra e può decidere se sedersi con il primo cliente, oppure da solo ad un nuovo tavolo. Supponiamo che siano $n_1 + \dots + n_k = n$ clienti seduti a k tavoli. Il cliente $n+1$-mo può scegliersi se unirsi al tavolo k già occupato, con una probabilità proporzionale al numero n_k di clienti già seduti a tale tavolo, oppure può sedersi ad un nuovo tavolo. Quest'ultima opzione può realizzarsi con una probabilità proporzionale a α.

Quindi riprendendo quanto si vede nella figura (4.3), il tavolo 1 ha una probabilità di venire scelto pari a $\frac{n_1}{\alpha+n}$, il secondo tavolo $\frac{n_2}{\alpha+n}$, ..., il tavolo k una probabilità di $\frac{n_k}{\alpha+n}$, mentre un nuovo tavolo ha probabilità di essere scelto pari a $\frac{\alpha}{\alpha+n}$.

La distribuzione tra le partizioni rimane quella appena descritta. Il fatto che i ristoranti cinesi abbiano tavoli rotondi è un aspetto fondamentale perché non solo questo processo vale tra le partizioni, ma definisce anche

la distribuzione delle permutazioni dove ogni tavolo corrisponde ad un ciclo di permutazioni.

A questo punto, possiamo considerare la distribuzione del numero di cluster (m) tra le n osservazioni. Per $i \geq 1$, l'osservazione θ_i assume un nuovo valore (creando un nuovo gruppo) con probabilità $\frac{\alpha}{\alpha+i-1}$ indipendentemente dal numero di cluster presente tra i θ precedenti. Si può dimostrare quanto valgono in modo approssimabile media e varianza della distribuzione:

$$\begin{aligned} E[m|n] &\simeq \alpha \log\left(1+\frac{n}{\alpha}\right) \qquad N, \alpha \gg 0 \\ Var[m|n] &\simeq \alpha \log\left(1+\frac{n}{\alpha}\right) \qquad n > \alpha \gg 0 \end{aligned} \tag{4.7}$$

Si può notare che il numero di cluster aumenta in scala logaritmica con il numero di osservazioni. Dato il fenomeno che "i ricchi si arricchiscono", questa lenta crescita è in linea, quindi ci aspettiamo cluster grandi dove il numero di cluster m è minore di n, numero di osservazioni. Il parametro α controlla i gruppi in maniera diretta. Infatti, un α grande indica un gran numero di gruppi a priori.

4.4 Indian buffet process

Nel clustering, ogni osservazione appartiene ad uno ed un solo gruppo. Ci sono, però, diversi problemi dove i cluster sono sovrapposti. Quindi, una singola osservazione può appartenere a più cluster. Gli elementi che appartengono alla famiglia di questi insiemi, vengono detti blocchi.

Per codificare questi elementi si procede costruendo una matrice binaria $\mathbf{Z}$, dove l'elemento della matrice $z_{ik} = 1$ se e solo se i appartiene al blocco k.

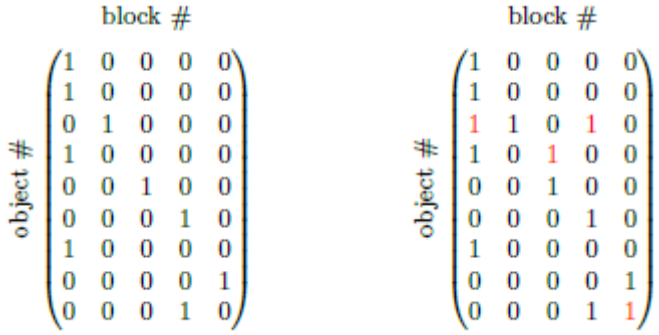


Figura 4.4: A sinistra un esempio di partizione, cioè un caso particolare della famiglia di insiemi, dove gli elementi possono appartenere ad un solo gruppo. A destra una famiglia di insiemi, dove gli elementi in rosso appartengono a più di un gruppo [27].

L'*Indian buffet process* rappresenta una generalizzazione del processo del ristorante cinese. In particolare, è chiamata così la distribuzione relativa alle matrici binarie descritte dal seguente algoritmo:

1. Per $n = 1, 2, \ldots,$

   (1) inserire n in ciascun blocco Ψ_k separatamente con probabilità $\frac{|\Psi_k|}{n}$

   (2) creare dei nuovi blocchi con $Pois\left(\frac{\alpha}{n}\right)$, ognuno contenente solo n

2. Ottenere una matrice *left-order* il cui output risultante permette di ottenere la matrice **Z**,

dove la matrice **Z** è una matrice binaria casuale. La legge di **Z** dovrebbe essere invariante sotto permutazioni delle righe (*row-exchangeable*). La matrice generata nei punti (1) e (2) dell'algoritmo viene ordinata in modo unico che elimini l'ordine, ma questo non elimina la dipendenza della distribuzione. Per farlo viene utilizzata una distribuzione su matrici di *classe equivalente*. Questo metodo viene chiamato *left-ordering*:

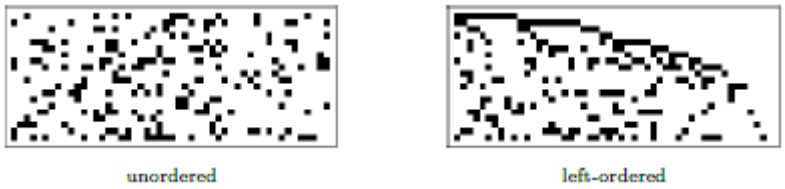


Figura 4.5: Esempio di *left-ordering* [27].

In una matrice randomica campionata da questo tipo di processo, sia le righe che le colonne sono scambiabili. Questo indica che la matrice **Z** è invariante a riordinamenti di righe o colonne. La somma delle righe ha sempre la stessa distribuzione $Pois(\alpha)$ marginalmente e questo è sempre vero nonostante la *left-ordering*. Tuttavia, prima dell'utilizzo di questa metodologia non è verificata la scambiabilità tra le righe.

4.5 Chinese Restaurant Franchise

Questo processo è l'analogo del processo del ristorante cinese, ma in ambito di processi gerarchici. Ora la metafora del ristorante cinese viene estesa a più ristoranti che condividono una serie di piatti.

Nel *Chinese Restaurant Franchise* (CRF) abbiamo un franchise di ristoranti con un menù condiviso tra essi, con un ristorante per ogni j. I clienti arrivano al j-esimo ristorante e decidono dove sedersi con lo stesso meccanismo del CRP. Questo procedimento è indipendente tra i vari ristoranti. Quindi, multipli tavoli in multipli ristoranti possono servire lo stesso piatto dal menù condiviso, consentendo così l'associazione tra i diversi ristoranti.

Il primo cliente arriverà ad uno dei j ristoranti, si siederà ad un tavolo e sceglierà quale piatto ordinare. Tutti i clienti che sceglieranno di sedersi allo stesso tavolo condivideranno quel piatto. Pertanto, i clienti sono le osservazioni x_{ji} e i ristoranti sono i diversi cluster. Allo stesso modo, i piatti del menù condiviso tra i clienti corrispondono a $\theta_1^*, \ldots, \theta_k^*$.

Dobbiamo introdurre una variabile indicatrice ψ_{jt} in modo da rappresentare quale piatto viene servito al tavolo t del ristorante j. Osserviamo che con questa costruzione abbiamo che ogni θ_{ji} è associato con un ψ_{jt}.

In questo processo, il cliente i entra nel ristorante j si siede al tavolo t_{ij}, dove a questo tavolo si serve il piatto θ_{ji}. Più esplicitamente, un cliente x_{ji} entra nel ristorante j e sceglie di sedersi al tavolo t_{ji} dove sono già presenti n_{jt} clienti e condivide con loro il piatto ψ_{jt}. Questo avviene con una probabilità pari a $\frac{n_{jt}}{+i-1}$. Può, però, anche scegliere di sedersi ad un nuovo tavolo t^{new} con probabilità $\frac{\alpha}{\alpha+i-1}$ e ordinare un nuovo piatto (cluster) ψ_{jt}^{new}.

Il partizionamento è descritto in base a come avviene la distribuzione dei clienti che ripercorre ciò che è già stato detto nel CRP, e nel dettaglio ciò viene descritto dalla distribuzione condizionata dei θ_{ji}:

$$\theta_{ji}|\theta_{j1},\dots,\theta_{j,i-1},,G_0 \sim \sum_{t=1}^{T_j} \frac{n_{jt}}{+i-1}\delta_{\psi_{jt}} + \frac{\alpha}{\alpha+i-1}G_0 \tag{4.8}$$

dove G_j è stata integrata. Questa è una mistura dove il termine di destra fornisce la probabilità di estrazione tenendo conto delle corrispondenti proporzioni di mistura. Un esempio di CRF si può vedere nella Figura 4.6.

La scelta del nuovo piatto fatta dal cliente nel ristorante j al tavolo t^{new} viene definita dal secondo processo di Dirichlet. Difatti, potrà scegliere se ordinare un piatto già ordinato θ_k^* da m_k tavoli degli altri ristoranti con probabilità $\frac{m_k}{\sum_{s=1}^k m_s+\gamma}$, oppure potrà ordinare un nuovo piatto θ_k^{new}, dove $\theta_k^{new} \sim H$, con probabilità $\frac{\gamma}{\sum_{s=1}^k m_s+\gamma}$.

In questo modo permettiamo che i cluster siano condivisi e possiamo definire la distribuzione condizionata della variabili indicatrice ψ_{ji} associata a θ_{ji}:

$$\psi^{new}|\boldsymbol{\psi},\gamma,H \sim \sum_{k=1}^{K} \frac{m_k}{\sum_{s=1}^k m_s+\gamma}\delta_{\theta_k^*} + \frac{\gamma}{\sum_{s=1}^k m_s+\gamma}H \tag{4.9}$$

4.6 Gaussian process

Il processo gaussiano è una delle distribuzioni più semplici quando stiamo trattando funzioni continue e, per questo, anche una delle più utilizzate.

Definiamo $\mathbf{T}$ lo spazio delle funzioni da un insieme $S \subset \mathbf{R}^d :\to \mathbf{R}$ e Θ come un elemento casuale di $\mathbf{T}$. Fissando un punto $s \in S$, $\Theta(s)$ è una variabile casuale in $\mathbf{R}$. Possiamo generalizzare quanto appena detto per più punti. Siano $s_1,\dots,s_n \in S$, n punti tali che $\Theta(s_1),\dots,\Theta(s_n)$ è un vettore casuale in $\mathbf{R}^n$.

Definizione 6 (Gaussian process)**.** *Sia μ una misura di probabilità sullo spazio $\mathbf{T}$. Le distribuzioni*

$$\mu_{s_1,\dots,s_n} := \Phi(\Theta(s_1),\dots,\Theta(s_n)) \tag{4.10}$$

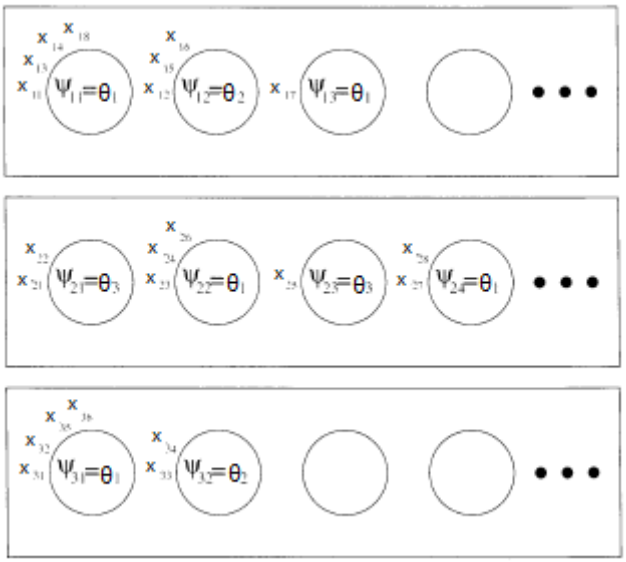


Figura 4.6: Un esempio di *Chinese Restaurant Franchise*. Ogni ristorante è rappresentato da un rettangolo, i clienti sono rappresentati da x_{ij} e i tavoli a cui sono seduti hanno forma circolare. Ad ogni tavolo viene servito un piatto del menù globale. La variabile ψ_{jt} rappresenta l'indicatore specifico di ciascun tavolo con il piatto corrispondente. Il cliente si siede al tavolo a cui è stato assegnato nell'equazione 4.8 [38].

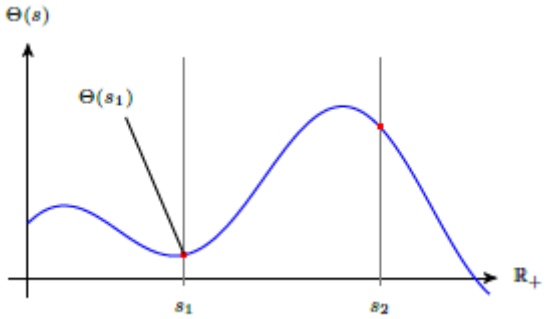


Figura 4.7: Una funzione casuale $\Theta : \mathbf{R} \rightarrow \mathbf{R}$ definisce uno scalare casuale $\Theta(s)$ per ogni punto $s \in \mathbf{R}$ [27].

definite da μ sono dette marginali finito-dimensionali *(o* distribuzioni finito-dimensionali*) di μ. Se $\mu_{s_1,\ldots,s_n}$ è una normale n-dimensionale per ogni insieme finito di punti $s_1,\ldots,s_n \in S$, allora μ è definito* Gaussian process *(GP) su* **T**.

Questa definizione comporta delle assunzioni implicite:

1. μ su **T** esiste;
2. il processo è unicamente definito dalle distribuzioni finito-marginali, che in generale non è né ovvio né universalmente vero.

Il processo gaussiano può essere utilizzato come prior, ma solo se è poi possibile riuscire a trovare la posterior corrispondente.

Possiamo vedere il processo gaussiano come mistura di v.c. $Normali$. Generiamo una mistura gaussiana infinita, come in questo esempio:

$$\begin{aligned} P &\sim DP(N(0,5)) \\ \mu_i &\sim P \qquad \text{per} \quad i = 1,\ldots,n \\ X_i|\mu_i &\sim N(\mu_i,1) \qquad \text{per} \quad i = 1,\ldots,n \end{aligned} \tag{4.11}$$

Permette di trovare una prior non parametrica continua:

$$p(X|P) = \sum_{i=1}^{\infty} \pi_i \phi(X - \mu_i) \tag{4.12}$$

dove π_i è la proporzione di mistura tale che $\sum_{i=1}^{\infty} \pi_i = 1$ e ϕ rappresenta la funzione di densità di probabilità normale.

Questo tipo di modello è molto utile perché può essere combinato con il processo di Dirichlet in modo da ottenere il cosiddetto DPGP (*Dirichlet Process Gaussian Process mixtures*). Tale mistura viene utilizzata quando si hanno problemi di apprendimento permanente e di rilevamento dal momento che è possibile pianificare delle sequenze di misurazione che ottimizzano i valori dei dati futuri [39].

4.6.1 Supersmooth & ordinary smooth

Parlando di misture di DP possiamo avere delle misture di kernel normali. In questa occasione specifica dobbiamo fare una distinzione in due casi:

1. il caso *supersmooth* dove la vera densità è una mistura di normali con deviazione standard compresa tra due valori positivi.
2. il caso *ordinary smooth* dove la vera densità è continua e differenziabile due volte, ma non necessita di essere una mistura di normali.

La prior di mistura di Dirichlet usata nel primo caso restringe, appunto, la variabilità del kernel normale all'interno di due bande. E' necessaria l'assunzione di supporto compatto, ovvero chiuso e limitato, sia per la distribuzione con mistura, sia per la distribuzione base di tali DP.

Una mistura normale con tasso di convergenza ϵ può essere approssimata tramite una mistura finita di normali con soli punti del supporto tali che si abbia una velocità di convergenza pari a $O\left(log\frac{1}{\epsilon}\right)$. L'assunzione di compattezza reca meno preoccupazioni circa la lenta crescita della funzione logaritmica, infatti il tasso di convergenza è molto simile a quello parametrico.

Per l'*ordinary smooth*, si richiede che il parametro di scala sia vicino a zero con una probabilità sufficientemente alta, così che una sequenza di prior possa essere costruita scalando una prior fissa tramite qualche sequenza σ_n.

5
Applicazione: dati Twitter

In questo capitolo ci occuperemo di applicare un modello gerarchico bayesiano non parametrico basato sul processo di Dirichlet a dei dati raccolti da Twitter. Per questa analisi sono state utilizzate sia tecniche bayesiane non parametriche, che metodi gerarchici che non si basano su queste metodologie così da poter effettuare un confronto.

Presenteremo inizialmente un esempio di applicazione nell'analisi testuale per introdurre il modello usato da Blei (2003) [5] e utilizzato successivamente nella applicazione ai dati Twitter.

5.1
Esempio di applicazione nell'analisi testuale

Quanto visto nei precedenti capitoli tratta l'analisi dei gruppi solamente a livello teorico, ma non dice nulla su come avviene a livello applicativo. Vediamo, quindi, come la proprietà di *clustering* viene applicata su dati relativi all'analisi testuale.

In particolare, vedremo come sia possibile ottenere cluster separati, ma che rimangano in collegamento tra loro.

Quando eseguiamo *textual analysis* troviamo un ampio uso dei modelli bayesiani. Questi modelli vengono chiamati *topic models* e si occupano della classificazione di documenti e testi. Essi sono una classe di tecniche di *text mining* non supervisionate. Esistono, infatti, diversi algoritmi di *topic modeling*, tra i quali il più diffuso e applicato è noto come *Latent Dirichlet Allocation* (LDA) [9].

In generale, questi modelli si basano sull'idea di individuare quali siano gli argomenti all'interno di un documento necessari per descriverlo. Si occupano quindi di trovare dei cluster ipotetici all'interno di ogni testo. I principali argomenti che descrivono il documento vengono chiamati *topic*. Un *topic*, come scrisse Benjamin Schmidt (2012) [34], ha le seguenti proprietà:

> First, it is coherent: a topic is a set of words that all tend to appear together, and will therefore have a number of things in common. Second, it is stable: if a topic appears at the same rate in two different types of documents, it means essentially the same thing in both.

I documenti sono visti come un insieme di parole dove non è importante l'ordine con cui compaiono, cioè vengono detti *bag of words* ed è verificata la proprietà di scambiabilità tra le parole. Questa proprietà giustifica l'utilizzo di un approccio bayesiano. I *topic* possono appartenere anche a documenti differenti ed essere condivisi, la raccolta di documenti viene chiamata *corpus*. Possiamo, quindi, utilizzare un modello basato sul processo di Dirichlet, o meglio, un modello gerarchico che permetta la condivisione

tra *topic* all'interno del *corpus*. I modelli gerarchici non sono altro che l'estensione di un modello parametrico verso un numero infinito di cluster. Tramite il DP possiamo applicare questi tipi di modelli anche nell'analisi testuale.

La LDA fu sviluppata da David Blei (2003) ed è un modello gerarchico bayesiano a tre livelli di gerarchia, dove ogni documento della collezione è modellato come una mistura finita su un set di *topic* che corrispondono alle componenti di mistura. Ogni argomento a sua volta è una mistura infinita su un insieme di probabilità riferite ad ogni *topic*. Nell'analisi testuale, queste probabilità producono un'esplicita rappresentazione del documento [5]. Quindi, le proporzioni di mistura sono estrazioni da una distribuzione a priori dei *topic* del documento e, date tali proporzioni, possiamo notare che ogni estrazione indipendente dal modello di mistura corrisponde ad una parola del documento.

Entrando più nello specifico, consideriamo J documenti che costituiscono il *corpus* e K *topic* per la collezione di testi in analisi. Le parole sono indicate con w_{ji}, dove $i = 1,\dots,n_j$ e $j = 1,\dots,J$, e in totale abbiamo n_j parole per ogni documento. Come fatto anche in precedenza, associamo ad ogni parola una variabile indicatrice z_{ji} che, quando la parola w_{ji} appartiene al *topic* k, assume valore $z_{ji} = k$. Passando a considerare le distribuzioni, possiamo scrivere un modello dove le parole w_{ji} hanno una distribuzione a priori $F(\varphi_{z_{ji}})$ e i *topic* hanno una prior $Multinomiale$ con parametri θ_j. Possiamo considerare che N rappresenti la numerosità dell'insieme per $i = 1,\dots,n_j$ di parole di ogni documento e che si distribuisca come una $Pois(\xi)$, dove ξ assumerà un certo valore. Inoltre, dato che $F(\varphi_{z_{ji}})$ è la distribuzione delle parole nel *topic* identificato con $z_{ji} = k$, allora possiamo scrivere che $F(\varphi_{z_{ji}})$ sarà anch'essa una $Multinomiale$ di parametro φ_k. Ovviamente avremo delle distribuzioni a priori di Dirichlet con iperparametri rispettivamente pari a $\boldsymbol{\alpha} = (\alpha_1,\dots,\alpha_K)$ e $\boldsymbol{\beta} = (\beta_1,\dots,\beta_V)$, dove K è il numero di *topic* e V il numero di parole nel vocabolario, per i parametri θ_j e φ_k. Possiamo, allora, scrivere il modello completo come:

$$\begin{aligned} \theta_j|\boldsymbol{\alpha} &\sim Dir(\boldsymbol{\alpha}) & \varphi_k|\boldsymbol{\beta} &\sim Dir(\boldsymbol{\beta}) \\ z_{ji}|\theta_j &\sim Mult(\theta_j) & w_{ji}|z_{ji},\varphi_k &\sim Mult(\varphi_k) \end{aligned} \tag{5.1}$$

che graficamente corrisponde alla Figura 5.1, dove si riesce ad intuire bene i tre livelli di gerarchia su cui il modello LDA si fonda.

L'algoritmo della LDA è il seguente:

1. Estrarre $N \sim Pois(\xi)$
2. Estrarre $\theta_j \sim Dir(\boldsymbol{\alpha})$ per $j = 1,\dots,J$
3. Estrarre $\varphi_k \sim Dir(\boldsymbol{\beta})$ per $k = 1,\dots,K$
4. Per ognuna delle N parole, w_{ji} per $i = 1,\dots,n_j$:
   (a) estrarre un *topic* $z_{ji} \sim Mult(\theta_j)$
   (b) estrarre una parola $w_{ji} \sim Mult(\varphi_{z_{ji}})$

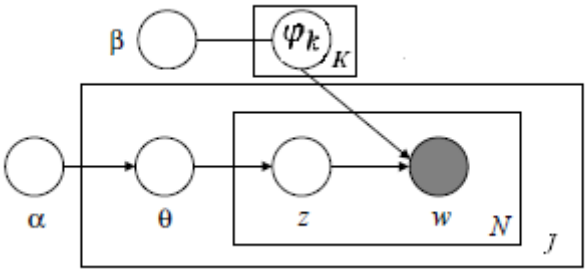


Figura 5.1: Rappresentazione grafica del modello di LDA. I cerchi sono i "piatti" che presentano replicazioni. La scatola esterna rappresenta i documenti, mentre la scatola interna rappresenta la scelta ripetuta di argomenti e parole all'interno di un documento [5].

Grazie a questa scrittura è facile notare come sia permessa la condivisione tra *topic* tra i diversi testi e come le parole siano provenienti da *topic* differenti. Bisogna fare un'osservazione in merito al numero K di *cluster* (*topic*) che sembrerebbe scelto a priori, ma grazie al fatto che ogni documento ha le proprie specifiche componenti di mistura possiamo non solo "far parlare i dati" e quindi derivare il numero ottimale di cluster, ma grazie all'estensione gerarchica permettere che ci sia condivisione delle stesse parole all'interno dei differenti *topic*.

Ulteriori approfondimenti e variazioni dell'algoritmo non verranno trattate, si rimanda a Blei (2003) [5].

5.2 Applicazione: Climate Change Twitter data

Prima di trattare l'analisi svolta, è utile capire le motivazioni delle scelta relativa alla tematica trattata e al social network utilizzato.

5.2.1 Climate Change e Twitter

Per l'analisi sono stati scelti dati associati al cambiamento climatico. Come possiamo leggere dal sito del WWF [1]:

> Il cambiamento climatico è una realtà e sta già provocando impatti e fenomeni di frequenza e intensità mai visti nella storia umana e con essi sofferenze, perdita di vite, sconvolgimento degli ecosistemi e della ricchezza di biodiversità che sostengono la nostra vita.

E' una tematica molto sentita dalla popolazione. Di notevole rilevanza le ultime campagne di sensibilizzazione della giovane attivista *Greta Thumberg* che con il suo motto "FridaysForFuture" è riuscita a far manifestare migliaia di persone al fine di combattere il cambiamento climatico.

[1] Se si vuole approfondire, si può leggere l'articolo al seguente link: `https://www.wwf.it/il_pianeta/cambiamenti_climatici/`

Tutti noi stiamo risentendo di questi cambiamenti dovuti a nostre azioni sbagliate negli anni passati. E' interessante andare a studiare come le persone percepiscono il problema e analizzarne i pensieri attraverso ciò che scrivono sui social network. In particolare, si è scelto di utilizzare Twitter.

Twitter [2] è un social network gratuito e microblogging in quanto consente alle persone di comunicare tra loro tramite dei messaggi di testo chiamati *tweets* che abbiano una lunghezza massima di 280 caratteri (eccetto per cinese, giapponese e coreano). Fu creato da Jack Dorsey a marzo dell'anno 2006 e raggiunse presto popolarità a livello mondiale. Su Twitter è possibile seguire dei profili, cioè far sì che le notifiche inerenti a quella persona siano visibili sulla home page, e venire seguiti da altrettante persone, dette *followers*. I *tweet* seguono un ordine cronologico inverso nella visualizzazione sulla pagina. E' possibile, inoltre, condividere informazioni utilizzando *tag* (@) e *hastag* (#). Gli *hastag* servono per etichettare parole chiave all'interno del testo del *tweet* e determinano in questo modo gli argomenti trattati. I *tag*, invece, servono per coinvolgere altri profili (utenti) nella conversazione. Gli *hastag* possono diventare virali e raggiungere ampia visibilità in tutto il mondo. E' ciò che è successo con l'*hastag* "climatechange" che a livello mondiale conta una visualizzazione di quasi 1 milione e mezzo di persone all'ora.

5.2.2 Dataset e analisi preliminari

Grazie alla possibilità di scaricare dati da Twitter, tramite delle credenziali API, è possibile scaricare fino ad un massimo di 5000 *tweet*. Il dataset è costituito da 4987 osservazioni (righe) e 73 variabili (colonne), originariamente le colonne erano 85, ne sono state rimosse 12 poiché costituite interamente da valori mancanti. I dati sono relativi al cambiamento climatico, difatti la chiave per scaricare questi *tweet* è stato l'*hastag* "#climatechange". I *tweet* raccolti fanno riferimento ai giorni 23/07/2019 e 22/07/2019 dove abbiamo 4802 osservazioni relative al primo giorno e 185 al secondo. Twitter permette di scaricare solo dati relativi massimo ad una settimana precedente andando a ritroso coi giorni, arrivato al limite l'algoritmo si stoppa per le nuove politiche di privacy a cui i social network sono sottoposti.

Viene eseguita un'analisi preliminare che tenga conto di un'opportuna pulizia del dataset, difatti vengono rimossi caratteri speciali che non vengono codificati correttamente da R, software utilizzato per tutto il processo di analisi. Sono state anche rimosse le cosiddette *stop words*, ovvero tutte quelle parole come pronomi, articoli, proposizioni, ... che sarebbero risultate molto frequenti e poco informative. Inoltre, successivamente alla rimozione di eventuali url, spazi bianchi e numeri, tutte le parole sono state trasformate in minuscolo, così da non avere problemi di *case sensitive*.

Il vocabolario risulta composto da 12016 parole. Abbiamo condotto un'analisi delle frequenze delle parole senza penalizzazione e un'analisi con penalizzazione tramite l'utilizzo della funzione peso *tf-idf* (*term frequency - in-*

[2] Per chi non conoscesse Twitter e voglia saperne di più, può utilizzare il seguente link `https://en.wikipedia.org/wiki/Twitter`

verse document frequency) ponendole a confronto. La seconda metodologia associa una maggiore importanza a termini poco frequenti nel documento.

La *tf-idf* dà ad ogni termine un'importanza che aumenta in modo proporzionale col numero di volte in cui il termine è incluso nel documento, ma penalizza in modo inversamente proporzionale l'importanza in base a quante volte il termine è contenuto nel *corpus*. Siano t_i una parola appartenente al documento d_j, n_{ij} il numero di occorrenze del termine e D il *corpus* [5]. Allora, indicando con #{•} la cardinalità di •, quindi la numerosità di termini presenti nello specifico documento, possiamo scrivere:

$$tf_{ij} = \frac{n_{ij}}{\#\{d_j\}} \tag{5.2}$$

che rappresenta le frequenze dei termini in ogni documento. Il secondo elemento che introduciamo serve, invece, per misurare l'importanza del termine nella collezione di documenti:

$$idf_i = \log \frac{\#\{D\}}{\#\{d : t_i \in d\}} \tag{5.3}$$

al denominatore troviamo la numerosità di documenti che contengono la parola t_i. Aggregando i due elementi possiamo riportare come avviene il calcolo dei pesi per ogni parola:

$$(tf - idf)_{ij} = tf_{ij} \times idf_i \tag{5.4}$$

Questa tecnica permette di ridurre il numero di parole che stiamo considerando nell'analisi, perché alcuni termini avranno associato un peso molto basso.

In questo modo il vocabolario si è ridotto ulteriormente ed è stato possibile, scegliendo una frequenza minima per le parole, andare a rappresentare due grafici a barre rappresentanti le parole con frequenza maggiore utilizzando e non utilizzando i pesi ottenuti con *tf-idf*. Nella Figura 5.2 è possibile constatare che il termine più importante è "climatechange", nonché termine chiave per scaricare i dati, mentre nella Figura 5.3 notiamo che è stato rimosso data la penalizzazione avvenuta tramite il peso. Siamo in grado anche di notare che nel secondo grafico abbiamo frequenze molto vicine e vi compaiono quasi tutti gli altri termini presenti nella prima figura.

Possiamo rappresentare le parole attraverso un *wordcloud* dove a parola più grande, corrisponde una frequenza associata maggiore. Come si può vedere dalla Figura 5.4, troviamo le stesse parole visualizzate nel grafico relativo alle frequenze, ma questa visualizzazione ha un impatto visivo molto più grande.

Avendo creato queste frequenze pesate, possiamo anche visualizzare per ogni parola di interesse, i termini che siano ad essa associata con un limite minimo di correlazione da noi scelto. Visualizzando, ad esempio, la parola "climate" ed i termini ad essa associata con una correlazione minima di 0.24, possiamo vederne i risultati nella Tabella 5.1.

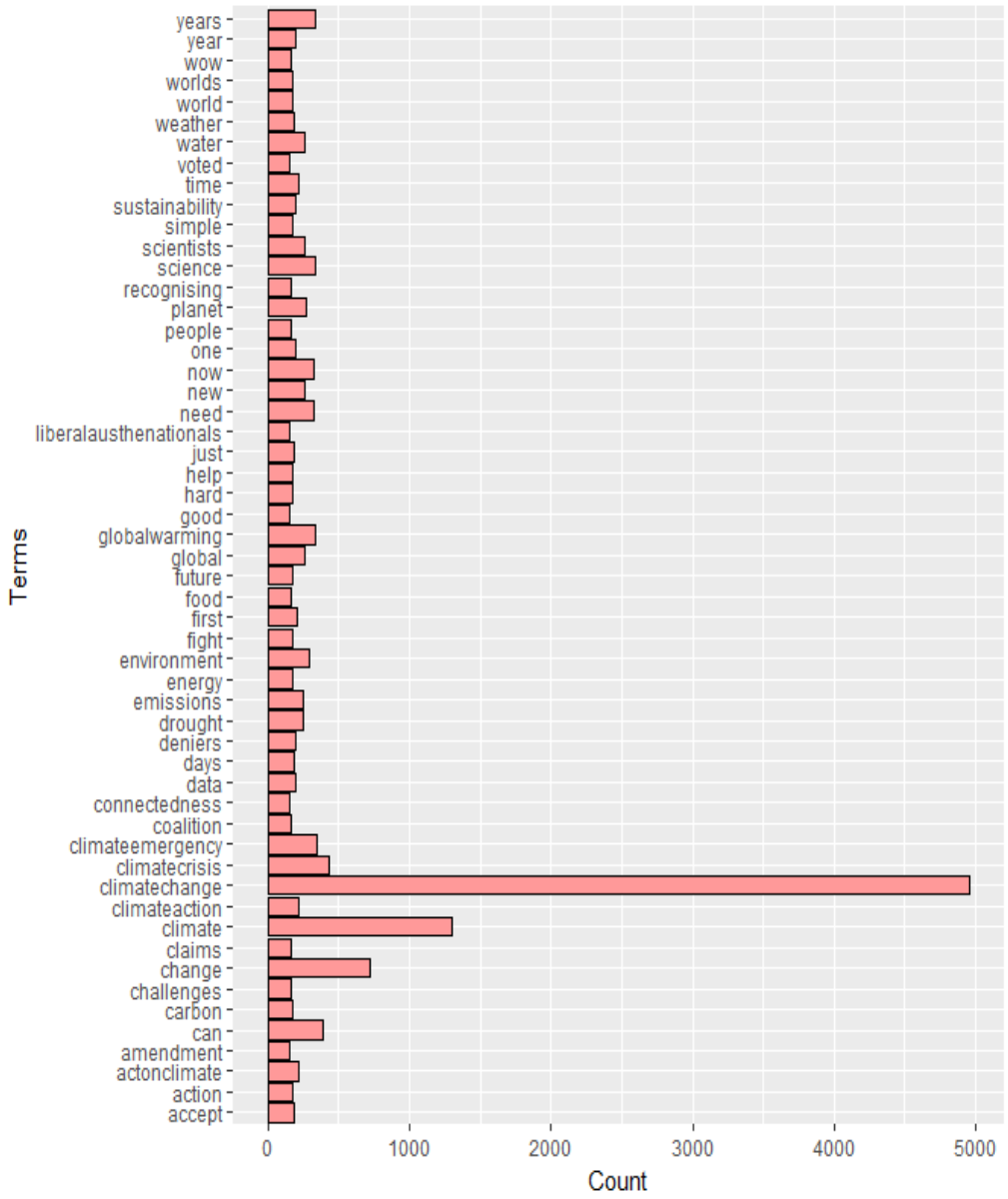


Figura 5.2: Grafico delle frequenze delle parole con frequenza minima 150.

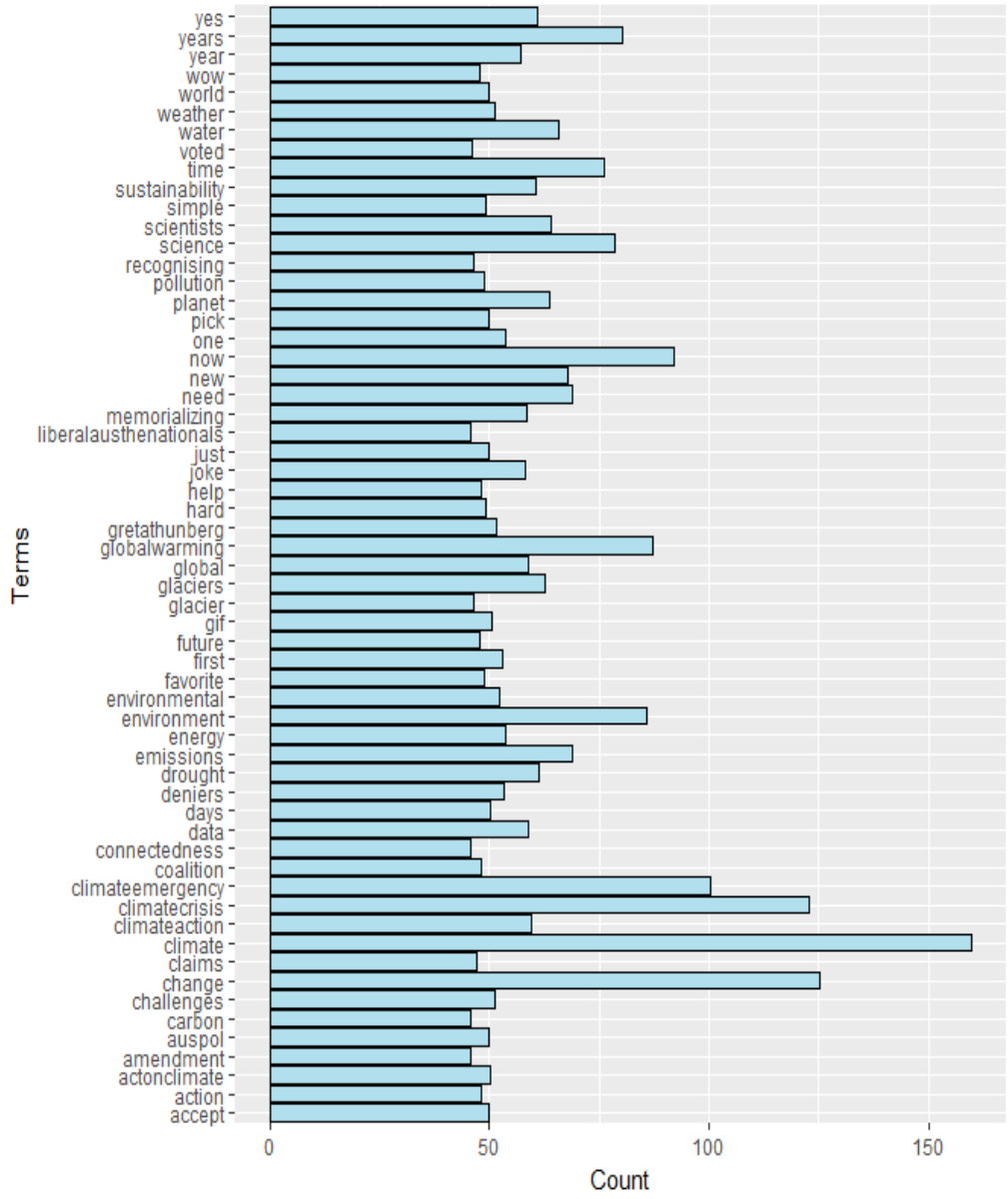


Figura 5.3: Grafico delle frequenze delle parole ottenute con pesi *tf-idf* e con frequenza minima 45.

Figura 5.4: Wordcloud delle parole con frequenza pesata.

Word	Correlation
change	0.55
deniers	0.26
amendment	0.25
claims	0.25
connectedness	0.25
liberalausthenationals	0.25
recognising	0.25
voted	0.25
accept	0.24

Tabella 5.1: Parole correlate con il termine “climate”.

5.2.3 Sentiment analysis

Un’ulteriore analisi sui *tweet* ottenuti, riguarda l’aspetto della *sentiment analysis*, ovvero andare a verificare la quantità di *tweet* con connotazione negativa e positiva che sono stati raccolti. Questo è utile perché ci fa già intuire quali siano i sentimenti delle persone riguardanti la tematica trattata.

Possiamo così ottenere la frase con connotazione più negativa:

> “As droughts, floods and hurricanes become more frequent, the world’s poorest people will be forced to starve or migrate. We risk a ’climate apartheid’ scenario where the wealthy pay to escape overheating and hunger while the rest of the world is left to suffer.”

quella con connotazione più positiva:

> "We need food systems that are sustainable, nutritious, inclusive and efficient. This means supporting a system that first protects the planet and second provides nutritious and diverse food."

ma anche una con connotazione neutrale:

> "This year will be a no-flight year for me. How about you?"

poiché ad ogni frase viene associato un punteggio positivo, negativo o neutrale in base ai termini trovati al suo interno. Possiamo anche essere interessati a verificare quali termini si attivino, cioè assumano un valore pari al numero di parole ad esse riferiti, quando leggiamo una determinata frase. Ad esempio, a queste tre frasi sopra riportare, le parole associate sono:

Frase	anger	anticipation	disgust	fear	joy	sadness	surprise	trust	negative	positive
Negativa	1	0	1	1	0	1	0	1	1	1
Positiva	0	1	0	0	1	1	0	4	1	6
Neutrale	0	0	0	0	0	0	0	0	0	0

Tabella 5.2: Termini attivati per ogni frase.

Nel dataset abbiamo un 25% di *tweet* negativi e un 41% di positivi, il rimanente è considerato neutrale.

Ulteriormente, è possibile effettuare una catalogazione dei termini che vengono utilizzati per costruire i *topic* nella nostra analisi. Riportiamo 10 parole tra le più positive e tra le più negative che risultano presenti nel nostro dataset nella Tabella 5.3.

Positive	Negative
benefits	disaster
accept	dangerously
aid	wars
sustainable	invasive
humanitarian	aggressively
opportunities	horrors
optimism	apocalypse
reliably	tragically
achieve	loses
productivity	fears

Tabella 5.3: Parole con connotazione positiva e negativa.

5.2.4 Clustering

In questa sezione ci occuperemo di effettuare una *cluster analysis* applicando un metodo gerarchico basato sul criterio di *Ward* e un cluster gerarchico utilizzando la funzione LDA di R[3].

Il primo metodo utilizzato è stato implementato per far emergere somiglianze e differenze rispetto ad un approccio bayesiano. Il cluster gerarchico è stato ottenuto togliendo dall'analisi i termini sparsi derivanti dalla matrice di dati pesati tramite *tf-idf*, quindi, per le analisi sono state utilizzate 255 parole. Ne risultano 7 cluster scelti tramite la visualizzazione del dendrogramma riportato in Figura 5.5 e 5.6. I gruppi hanno la seguente suddivisione:

Gruppo 1	Gruppo 2	Gruppo 3	Gruppo 4	Gruppo 5	Gruppo 6	Gruppo 7
16	1	16	181	14	14	13

Tabella 5.4: Numerosità termini associati ad ogni cluster.

Come si può notare, il secondo gruppo è composto da una singola parola "climatechange" che seppur pesata risulta scissa dalle altre, potremmo considerarla una parola a sé stante visto che compare in tutti i *tweet*. Gli altri gruppi racchiudono un quantitativo di parole similare eccetto per il quarto cluster che racchiude il 71% delle parole. Non sono stati creati ulteriori gruppi poiché i tagli del dendrogramma dovevano essere scelti molto bassi, ogni parola avrebbe formato un proprio gruppo. La difficoltà nell'applicazione di questa tecnica sta nella scelta del taglio da effettuare nel dendrogramma e, di conseguenza, nel decidere il numero degli ipotetici cluster presenti.

Procediamo allora con l'utilizzo del modello *Latent Dirichlet Allocation* dove come prior H delle parole in analisi abbiamo assunto una distribuzione di Dirichlet simmetrica con parametro pari a $\frac{1}{255} = 0.003921569$. Abbiamo eseguito un *Gibb-sampling* settando *burning period* pari a 4000, numero di iterazioni 2000 e *thin period* di 50.

Bisogna fare un'osservazione sul numero di cluster. Essendo un'analisi non parametrica, si avranno *potenzialmente* infiniti parametri e, di conseguenza, infiniti cluster. L'algoritmo produce inizialmente 144 cluster, dove il numero di *tweet* che condividono diverse parole varia da 2 a 2448. Tuttavia questa rappresentazione non risulta efficiente. La creazione di potenziali cluster avviene non tenendo conto del loro poter rappresentare la tematica di interesse. Si è così optato di riportare un numero inferiore di gruppi. I gruppi riportati risultano essere i più rappresentativi del cambiamento climatico, dato che contengono più parole al loro interno. Per determinare la numerosità ottimale di cluster da rappresentare ci si è basati su

[3]Tutto il codice per le analisi è reso disponibile sia nella sezione Appendice R & Python Code, sia nella repository github `https://github.com/AliceGiampino/MasterThesisHDP`

delle simulazioni in cui si è valutata la *perplexity*. Un valore basso di questo indicatore indica migliori performance di generalizzazione. Difatti, è direttamente proporzionale alla log-verosimiglianza, ma monotonicamente decrescente nella verosimiglianza dei dati di test.

$$perplexity(D_{test}) = \exp\left\{\frac{\sum_{j=1}^{J} log - likelihood(\boldsymbol{w})}{\sum_{j=1}^{J} N_j}\right\} \tag{5.5}$$

dove ricordiamo che N_j è il numero totale di parole del documento j e $\boldsymbol{w}$ rappresenta il vettore di parole.

Come possiamo vedere dalla Figura 5.7, il numero ottimale di cluster risulta essere 100.

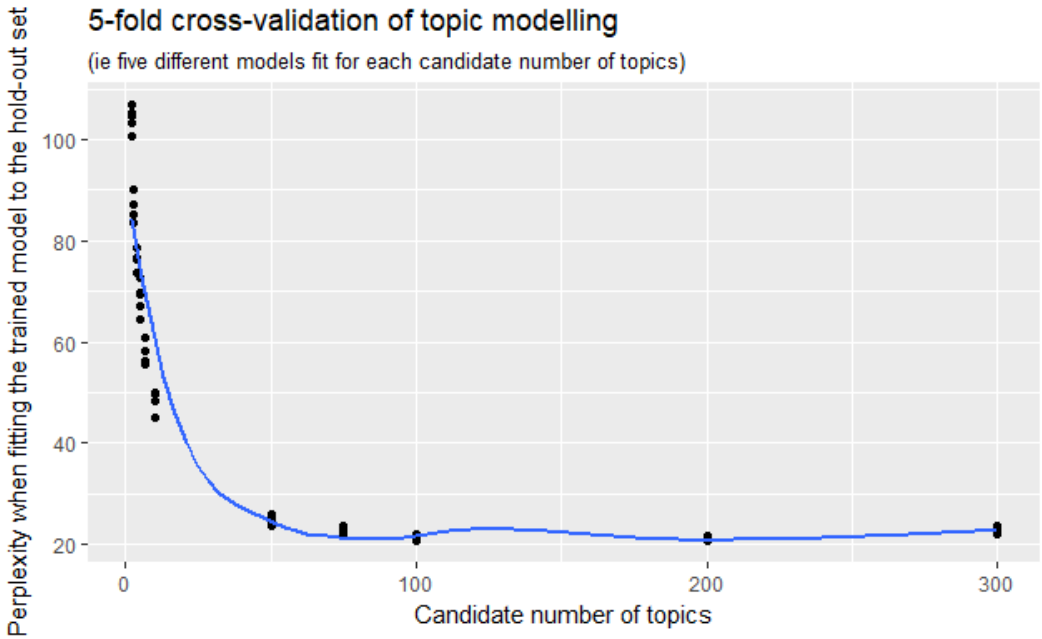


Figura 5.7: *Perplexity*.

Siccome stiamo lavorando con 255 parole questo valore per la numerosità dei gruppi risulta essere troppo ampio e poco informativo, dal momento che creeremmo gruppi dove i *topic* si somiglierebbero troppo o dove essi sarebbero composti da poche parole tutte sinonimi tra loro se non addirittura da singoli termini. Abbiamo, allora, provato a testare il numero ottimale di *topic* su differenti algoritmi dove si sceglie il numero di argomenti che minimizza "CaoJuan2009" [7] e "Arun2010" [3], mentre massimizza "Griffiths2004" [17] e "Deveaud2014" [10]. In questo caso, come emerge dalla Figura 5.9, il numero ottimale risulta essere 13. La metrica di "Griffiths2004" non giunge a convergenza e quella di "Arun2010", graficamente, non risulta informativa.

Dal momento che abbiamo scelto di effettuare le analisi con 13 cluster, vengono di seguito riportati i gruppi con le relative parole e le probabilità di estrazione:

Gruppo 1	
water	0.0420
just	0.0331
first	0.0322
solutions	0.0318
crisis	0.0318

Gruppo 2	
lost	0.0513
time	0.0413
glacier	0.0300
take	0.0296
becoming	0.0287

Gruppo 3	
air	0.0344
data	0.0323
think	0.0317
record	0.0301
great	0.0296

Gruppo 4	
emissions	0.0348
carbon	0.0341
tackling	0.0282
new	0.0262
actonclimate	0.0245

Gruppo 5	
years	0.0548
scientist	0.0543
ago	0.0516
science	0.0512
globalwarming	0.0507

Gruppo 6	
climate	0.0507
change	0.0491
weather	0.0349
data	0.0267
challenges	0.0254

Gruppo 7	
environment	0.0513
sustainability	0.0360
globalwarming	0.0315
pollution	0.0264
climateemergency	0.0249

Gruppo 8	
simple	0.0541
change	0.0535
drought	0.0538
claims	0.0535
accept	0.0535

Gruppo 9	
climate	0.0784
change	0.0624
help	0.0222
make	0.0162
future	0.0159

Gruppo 10	
fossil	0.0536
fuels	0.0498
industry	0.0452
keepitintheground	0.0438
oil	0.0340

Gruppo 11	
climatecrisis	0.540
now	0.0506
climateemergency	0.0319
gretathumberg	0.0310
action	0.0306

Gruppo 12	
climate	0.0363
heatwave	0.0266
arctic	0.0242
globalwarming	0.0215
world	0.0201

Gruppo 13	
india	0.0413
floods	0.0381
climateaction	0.0361
urgent	0.0340
bangladesh	0.0340

Nella Figura 5.8 possiamo vedere graficamente quanto riportato nelle tabelle dei gruppi.

Per comprendere al meglio la distribuzione delle parole nei *topic* e la loro condivisione è utile guardare il *wordcloud* in Figura 5.10.

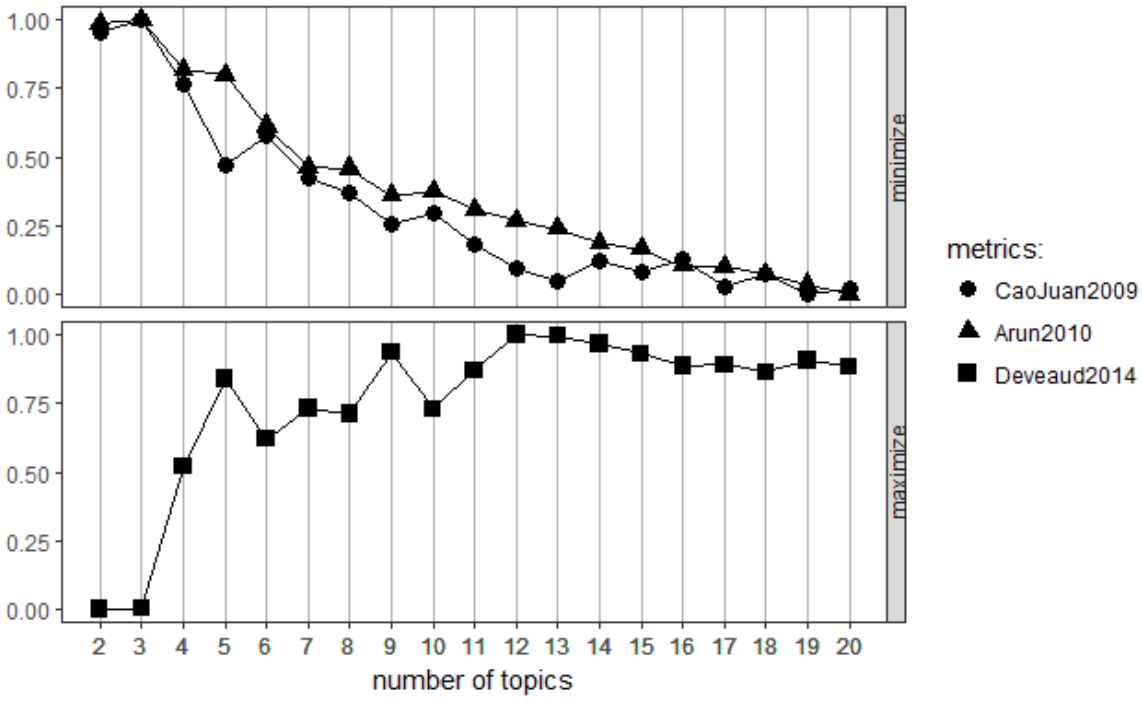


Figura 5.9: Numero di *topic* basato su 3 diverse metriche.

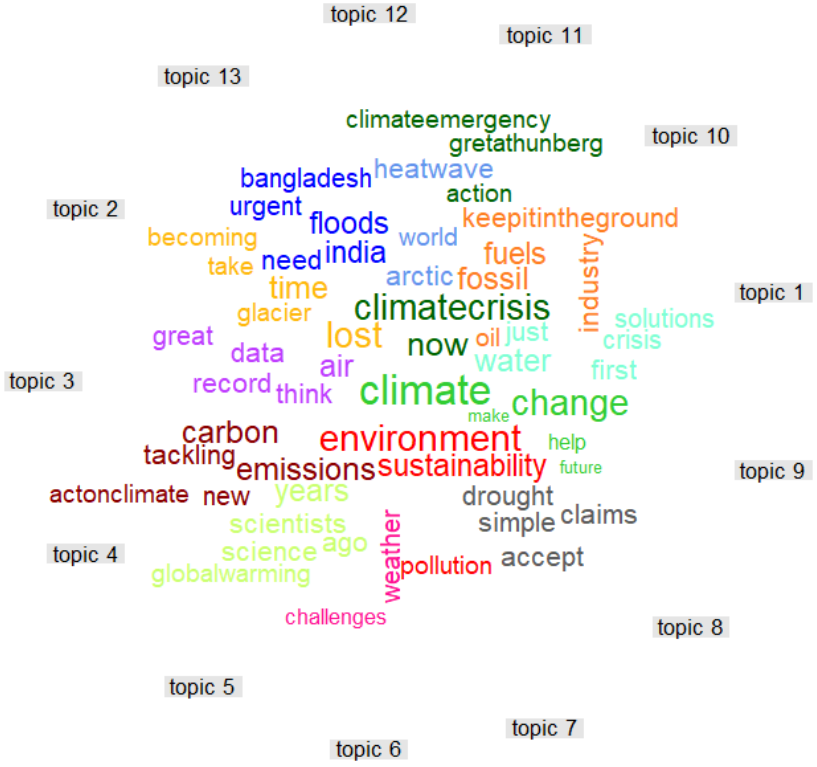


Figura 5.10: *Wordcloud* per *topic*.

5.3
Main findings and hints for future research

Osservando i *cluster* ottenuti, ci rendiamo conto che le parole che si suddividono nei diversi gruppi sembrino descrivere differenti *topic*, quali:

1. Crisi idrica.
2. Lo scadere del tempo per le azioni, come testimonia il rapido scioglimento dei ghiacciai.
3. Dati sull'inquinamento atmosferico.
4. Aumento delle emissioni.
5. Monito della comunità scientifica circa la gravità del problema.
6. Cambiamento climatico visto come sfida.
7. Emergenza climatica e sostenibilità ambientale.
8. Trovare semplici soluzioni alla siccità.
9. Agire per il futuro.
10. Combustibili fossili.
11. Attualità del cambiamento climatico.
12. Innalzamento delle temperature e scioglimento della calotta polare.
13. Alluvioni in India sempre più devastanti.

Inoltre, osserviamo che le parole in ciascun gruppo non solo sembrano coerenti tra loro, ma quelle riportate hanno la probabilità maggiore di essere estratte. Possiamo constatare la diversità rispetto al cluster gerarchico con metodo di *Ward*, dove non potevamo lasciarci guidare dai dati come in questo caso perché, come già fatto notare dal dendrogramma, sarebbe stata fuorviante la decisione sulla numerosità di cluster. In questo caso, invece, tramite le proporzioni di mistura il numero di cluster creato è quello ottimale per questa analisi. Difatti, i gruppi risultano essere formati in modo più coerente dai termini al loro interno.

Possiamo, inoltre, notare che questo approccio consente ai cluster di essere in collegamento tra di loro per mezzo della condivisione di alcuni termini. Nonostante questo legame tra i gruppi, osserviamo che sono caratterizzati da termini che mostrano aspetti differenti del cambiamento climatico, descrivendolo a 360°.

Supponendo di avere a disposizione un dataset composto da un numero maggiore di *tweet* relativi alla tematica di interesse, saremmo in grado di costruire un vocabolario più ampio al fine di descrivere il cambiamento climatico. Questo è uno dei possibili sviluppi di tale analisi che ha avuto

il semplice obiettivo di applicare in modo esemplificativo il modello LDA in una *textual analysis*. Ad ogni modo, questi algoritmi sono vivi argomenti di ricerca e vengono introdotte funzioni sempre più ottimizzate ed efficienti.

Il processo di Dirichlet è una delle colonne portanti della statistica bayesiana non parametrica e su esso si stanno sviluppando estensioni innovative molto utilizzate tra i più disparati campi di applicazione. Alcuni articoli dell'ultimo anno iniziano ad implementare tali modelli nell'area economica dove i metodi classici facevano da padrone. Risulta interessante sviluppare un'analisi siffatta al fine di indagare le performance di questi modelli nel campo econometrico, con lo scopo di superare problemi legati alle scelte arbitrarie.

Figura 5.5: Dendrogramma con suddivisione in 7 gruppi.

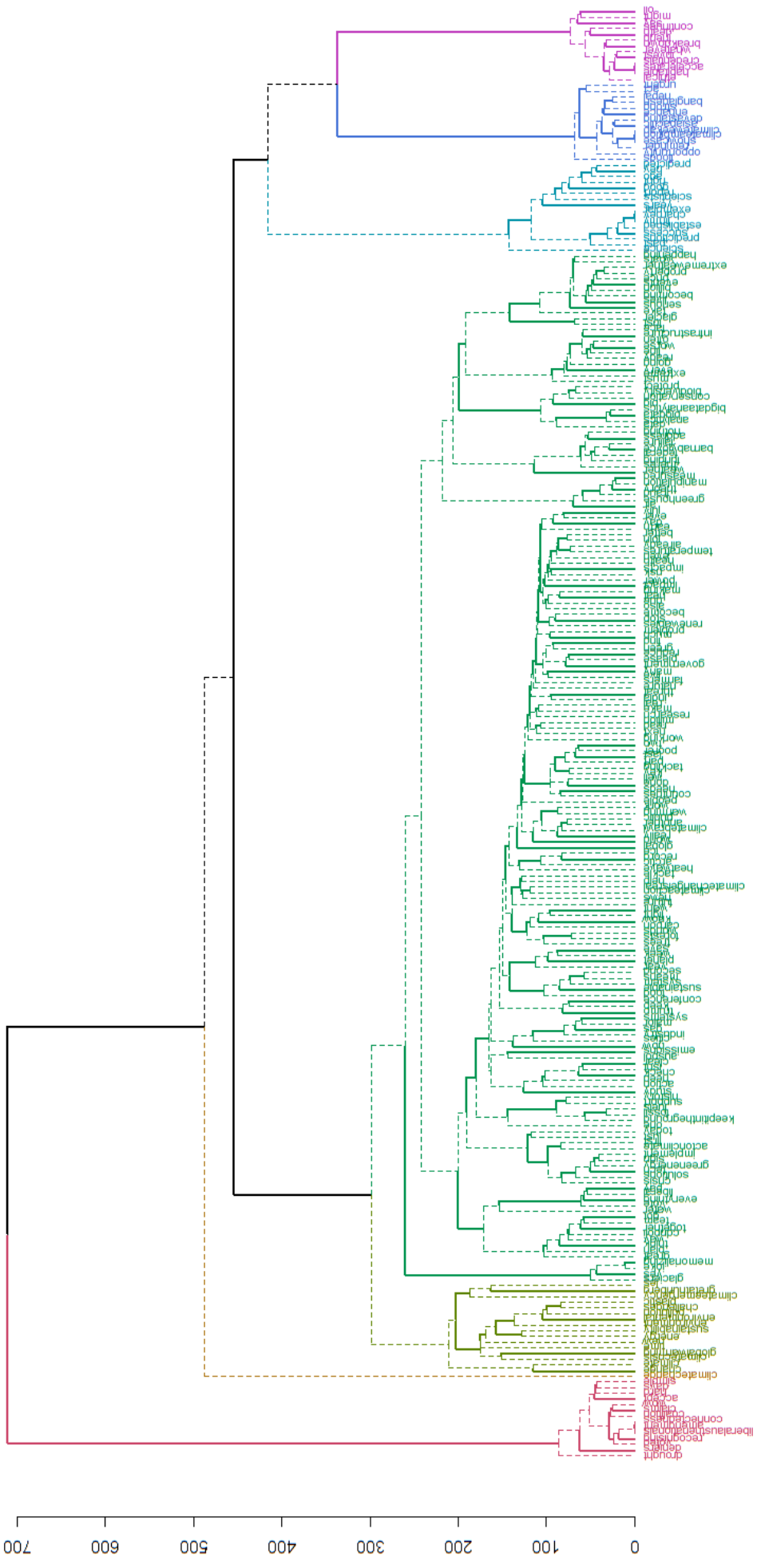


Figura 5.6: Dendrogramma con suddivisione visibile nei *branches* e relativi termini.

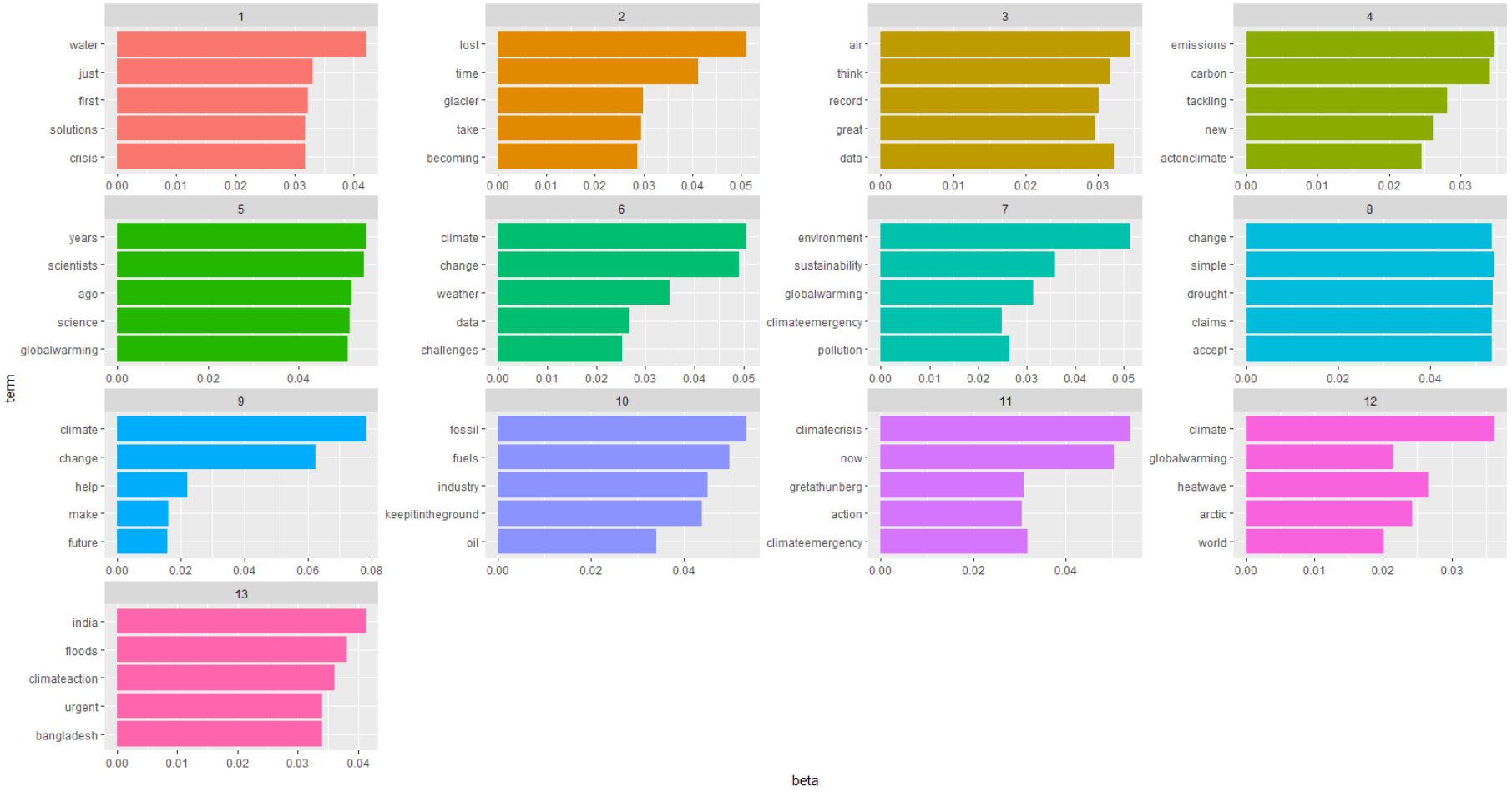


Figura 5.8: Frequenza delle parole in ciascun cluster (*topic*).

6
Conclusioni

Il processo di Dirichlet e, in particolare, i modelli di mistura con estensione gerarchica si sono dimostrati di non facile applicazione, ma hanno portato all'individuazione dei cluster più rappresentativi. Nonostante i gruppi coerenti che sono stati creati, ci sono alcune limitazioni che devono essere prese in considerazione.

Innanzitutto, come già anticipato, si potrebbe utilizzare un vocabolario molto più ampio che potrebbe arricchire in questo modo l'analisi. Inoltre, bisogna tener in considerazione il fatto che l'algoritmo da noi utilizzato è *data driven* e va a costruire il numero adeguato dei cluster in base alla quantità di dati a disposizione. Maggiore la numerosità delle osservazioni, maggiore il numero di cluster. Ciò può risultare problematico perché si potrebbero avere risultati distanti da ciò che ci si aspetta.

Un altro problema in cui si può imbattere riguarda la corretta implementazione del modello. Individuare il metodo corretto per implementarlo richiede tempo e attenzione, nonché la ricerca di analisi simili al fine di capire come gestire al meglio i parametri. Si può quindi affermare che il modello basato sul *Dirichlet Process* sia risultato più efficiente allo scopo di analizzare i dati sul cambiamento climatico ottenuti da Twitter. Ciononostante questa metodologia è ancora poco diffusa data la complessità della teoria sottostante.

Probabilmente negli anni avvenire ci sarà una maggiore valorizzazione ed utilizzo del processo di Dirichlet in campi come classificazione, genetica, econometria, apprendimento automatico e altri ancora.

Non sappiamo con certezza cosa ci attenderà nel futuro, ma di certo possiamo basarci sulle probabilità.

> *La teoria della probabilità è, in fondo, semplice buon senso tradotto in calcolo; ci fa valutare con esattezza ciò che una mente ragionevole sente per una sorta di istinto… E' degno di nota che questa scienza, nata a servizio dei giochi d'azzardo, sia diventata il più importante oggetto della conoscenza umana… Le più importanti questioni della vita sono, per la maggior parte, solo dei problemi di probabilità.*
>
> P.S. de Laplace

A
Distribuzioni

A.1
Distribuzione Poisson

Una variabile casuale Y ha distribuzione $Poisson$ con parametro λ, e si scrive sinteticamente $Y \sim Pois(\lambda)$ e funzione di densità di probabilità:

$$p_Y(y;\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma^2)}} e^{\frac{1}{2}\frac{(y-\mu)^2}{\sigma^2}} \tag{A.1}$$

I momenti di questa distribuzione sono

$$\begin{aligned} E(Y) &= \lambda \\ Var(Y) &= \lambda \end{aligned} \tag{A.2}$$

Dove $\lambda > 0$ e il supporto è $\mathbf{N}$.

A.2
Distribuzione Normale

Una variabile casuale Y ha distribuzione $Normale$ con parametri di forma μ, ($\mu \in \mathbf{R}$) e σ ($\sigma \in \mathbf{R}^+$), e si scrive sinteticamente $Y \sim N(\mu,\sigma)$ e funzione di densità di probabilità:

$$p_Y(y;\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma^2)}} e^{\frac{1}{2}\frac{(y-\mu)^2}{\sigma^2}} \tag{A.3}$$

I momenti di questa distribuzione sono

$$\begin{aligned} E(Y) &= \mu \\ Var(Y) &= \sigma^2 \end{aligned} \tag{A.4}$$

Importante la proprietà riproduttiva della v.c. $Normale$, cioè il prodotto di variabili indipendenti è ancora una $Normale$, ma con parametro media la somma dei parametri e come varianza la somma di varianze. La *Normale standard* ha media 0 e varianza unitaria.

A.3
Distribuzione Gamma

Una variabile casuale Y ha distribuzione $Gamma$ con parametri di forma α, ($\alpha \in \mathbf{R}^+$) e β ($\beta \in \mathbf{R}^+$), e si scrive sinteticamente $Y \sim Ga(\alpha,\beta)$ con supporto $\mathbf{R}^+$ e funzione di densità di probabilità:

$$p_Y(y;\mu,\sigma) = \frac{\beta}{\Gamma(\alpha)} y^{\alpha-1} e^{-\beta x} \tag{A.5}$$

I momenti di questa distribuzione sono

$$\begin{aligned} E(Y) &= \frac{\alpha}{\beta} \\ Var(Y) &= \frac{\alpha}{\beta^2} \end{aligned} \tag{A.6}$$

Importante la proprietà riproduttiva della v.c. $Gamma$, cioè il prodotto di variabili indipendenti è ancora una $Gamma$, ma con parametro di scala la somma dei parametri.

A.4
Distribuzione Beta

Una variabile casuale Y ha distribuzione $Beta$ con parametri di forma α ($\alpha > 0$) e β ($\beta > 0$), e si scrive sinteticamente $Y \sim Be(\alpha, \beta)$ se Y ha supporto $S_Y = [0, 1]$ e funzione di densità di probabilità:

$$p_Y(y; \alpha, \beta) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} y^{\alpha-1}(1-y)^{\beta-1} \tag{A.7}$$

per y $\in S_Y$ e $p_Y(y; \alpha, \beta) = 0$ altrove, dove $\Gamma()$ indica la funzione Gamma. Si mostra facilmente che:

$$\begin{aligned} E(Y) &= \frac{\alpha}{\alpha + \beta} \\ Var(Y) &= \frac{\alpha\beta}{(\alpha + \beta)^2(\alpha + \beta + 1)} \end{aligned} \tag{A.8}$$

A.5
Distribuzione Cauchy

Una variabile casuale Y ha distribuzione $Cauchy$ con parametri di forma α e β, e si scrive sinteticamente $Y \sim Cauchy(\alpha, \beta)$ se Y ha supporto $S_Y = \mathbf{R}$ e funzione di densità di probabilità:

$$p_Y(y; \alpha, \beta) = \frac{1}{\pi} y^{\alpha-1} \frac{\beta}{(y-\alpha)^2 + \beta^2} \tag{A.9}$$

Importante sottolineare che questa variabile casuale NON ha momenti finiti. Il rapporto $\frac{X}{Y}$ tra due variabili aleatorie indipendenti aventi distribuzione *Normale standard* $N(0, 1)$ segue la distribuzione di $Cauchy$ di parametri $(0, 1)$.

A.6
Distribuzione di Dirichlet

La distribuzione di $Dirichlet$ è una distribuzione di probabilità continua, dipendente da un vettore di numeri reali positivi α che generalizza la distribuzione $Beta$ al caso multivariato.

Una distribuzione di $Dirichlet$ di ordine $K \geq 2$ con parametri $\alpha_1, \ldots, \alpha_K > 0$ ha una funzione di densità di probabilità rispetto alla misura di Lebesgue sullo spazio euclideo R^{k-1}, data da:

$$p(x_1, \ldots, x_{K-1}, \alpha_1, \ldots, \alpha_K) = \frac{1}{B(\alpha)} \prod_{i=1}^{K} x_i^{\alpha-1} \tag{A.10}$$

per ogni $x_1, \ldots, x_{K-1} > 0$ tali che $x_1 + \cdots + x_{K-1} < 1$ e $X_K = 1\ x_1 \ldots x_{K-1}$, e la indicheremo con $Dir(\alpha_1, \ldots, \alpha_K)$. La costante di normalizzazione è la funzione Beta multinomiale, che può essere espressa in termini di funzione Gamma come:

$$B(\alpha) = \frac{\prod_{i=1}^{K} \Gamma(\alpha_i)}{\Gamma(\sum_{i=1}^{K} \alpha_i)} \tag{A.11}$$

con $\underline{\alpha} = (\alpha_1, \ldots, \alpha_K)$ [12].

Un caso particolare molto comune di questa distribuzione, è la distribuzione di Dirichlet simmetrica nella quale tutti gli elementi del vettore dei parametri $\underline{\alpha}$ hanno lo stesso valore. Quindi, la distribuzione può essere parametrizzata da un singolo valore α scalare, chiamato parametro di concentrazione.

Sia $\mathbf{X} = (X_1, \ldots, X_K) \sim Dir(\alpha)$, ovvero i primi $K-1$ elementi hanno densità descritta sopra, e sia $\alpha_0 = \sum_{i=1}^{K} \alpha_i$.

I momenti della distribuzione sono:

$$\begin{aligned} E(X_i) &= \frac{\alpha_i}{\alpha_0} \\ Var(X_i) &= \frac{\alpha_i(\alpha_0 - \alpha_i)}{\alpha_0^2(\alpha_0 + 1)} \end{aligned} \tag{A.12}$$

Le marginali hanno distribuzione:

$$X_i \sim Beta(\alpha_i, \alpha_0 - 1) \tag{A.13}$$

B
Approfondimenti

B.1
Kolmogorov's consistency theorem

Teorema 3. *Sia T un intervallo (temporale) e sia $n \in \mathbb{N}$. Per ogni $k \in \mathbb{N}$ e per ogni sequenza finita di tempi $t_1, \dots, t_k \in T$, sia $\nu_{t_1 \dots t_k}$ una misura di probabilità in $(\mathbb{R}^n)^k$. Supponiamo che queste misure soddisfino le due condizioni di consistenza:*

1. *per tutte le permutazioni π di $\{1, \dots, k\}$ e per ogni insieme misurabile $F_i \subseteq \mathbb{R}^n$,*

$$\nu_{t_{\pi(1)} \dots t_{\pi(k)}} \left(F_{\pi(1)} \times \cdots \times F_{\pi(k)} \right) = \nu_{t_1 \dots t_k} \left(F_1 \times \cdots \times F_k \right)$$

2. *per ogni insieme misurabile $F_i \subseteq \mathbb{R}^n$, $m \in \mathbb{N}$*

$$\nu_{t_1 \dots t_k} \left(F_1 \times \cdots \times F_k \right) = \nu_{t_1 \dots t_k, t_{k+1}, \dots, t_{k+m}} \left(F_1 \times \cdots \times F_k \times \underbrace{\mathbb{R}^n \times \cdots \times \mathbb{R}^n}_{m} \right).$$

Allora esiste uno spazio di probabilità $(\Omega, \mathscr{F}, \mathbb{P})$ e un processo stocastico $X : T \times \Omega \to \mathbb{R}^n$ tale che:

$$\nu_{t_1 \dots t_k} \left(F_1 \times \cdots \times F_k \right) = \mathbb{P} \left(X_{t_1} \in F_1, \dots, X_{t_k} \in F_k \right)$$

per tutti $t_i \in T$, $k \in \mathbb{N}$ e per ogni insieme misurabile $F_i \subseteq \mathbb{R}^n$, i.e. X ha $\nu_{t_1 \dots t_k}$ come sue distribuzioni finito-dimensionali relative ai tempi $t_1 \dots t_k$.

Osservazione: è sempre possibile prendere come spazio di probabilità $\Omega = (\mathbb{R}^n)^T$ e prendere per X il processo canonico $X : (t, Y) \mapsto Y_t$.

Pertanto, un modo alternativo di esprimere il teorema di estensione di Kolmogorov è che, data la validità delle condizioni di consistenza, esiste una misura (unica) ν su $(\mathbb{R}^n)^T$ con marginali $\nu_{t_1 \dots t_k}$ per ogni collezione finita di tempi $t_1 \dots t_k$. Questo teorema si applica quando T non è numerabile, ma il prezzo da pagare per questo livello di generalità è che la misura ν è definita solo sul prodotto σ-algebra di $(\mathbb{R}^n)^T$, che non è molto ampio [26].

B.2
Convergenza di martingale

Teorema 4. *Se $M = (M_n)_{n \geq 0}$ è una martingala a tempo discreto tale che*

$$\sup_{n \in N} E[M_n^+] < \infty \tag{B.1}$$

allora la successione $(M_n(\omega))_{n \geq 0}$ converge per $n \to \infty$.

B.3
Convergenza in distanza di Kolmogorov-Smirnov

Asserzione 6. *Prende il nome da Andrey Kolmogorov e Nikolai Smirnov e riguarda la convergenza tra una funzione di ripartizione empirica e quella vera, basata sulla distanza D_n. Sia F una qualsiasi funzione di distribuzione di probabilità su $\mathbb{R}$. La funzione di distribuzione di una variabile casuale X è $F(x) := F_X(x) := Pr(X \leq x)$. Per qualsiasi numero reale $x_1,\ldots,x_n$, la funzione di distribuzione empirica corrispondente è definita da $F_n(x) := \frac{1}{n}\sum_{j=1}^{n} 1_{x_j \leq x}$ dove $1_{x_j \leq x}$ se $x_j \leq x$ e 0 altrimenti. Siano $x_1,\ldots,x_n$ ordinati tali che $x_{(1)} \leq x_{(2)} \leq \cdots \leq x_{(n)}$. Quindi avremo $F_n(x) = 0$ per $x < x_{(1)}$, $F_n(x) = \frac{j}{n}$ per $x_{(j)} \leq x < x_{(j+1)}$ per ogni $j = 1,\ldots,n-1$ e $F_n(x) = 1$ per $x \geq x_{(n)}$. $x_1,\ldots,x_n$ saranno i valori osservati di alcune variabili casuali $X_1,\ldots,X_n$. Supponiamo di voler testare l'ipotesi H_0 che $X_1,\ldots,X_n$ sono i.i.d. con una determinata funzione di distribuzione fissa F. Sia F_n la funzione di distribuzione empirica basata su $X_1,\ldots,X_n$. Una forma della statistica del test di Kolmogorov per H_0 è*

$$D_n := \sup_x |(F_n - F)(x)|. \tag{B.2}$$

e rappresenta la distanza tra le due funzioni [30].

B.4
Teorema di Glivenko - Cantelli (1933)

Teorema 5. *Per ogni funzione di distribuzione F (continua o no) e corrispondenti funzioni di distribuzioni empiriche F_n basate su $X_1,\ldots,X_n$ i.i.d. (F), quasi certamente $D_n = \sup_x |(F_n - F)(x)| \to 0$ come $n \to \infty$, in altre parole, con probabilità 1, $F_n \to F$ uniformemente.*

Questo teorema implica che il test di Kolmogorov è ciò che viene chiamato consistente tra tutte le alternative:

Asserzione 7 (Corollario)**.** *Supponiamo che $X_1,\ldots,X_n,\ldots$ siano i.i.d F, tramite il test di Kolmogorov si è sottoposta a verifica l'ipotesi H_1 che siano i.i.d per qualche funzione di distribuzione $G \neq F$ (dove F e G non devono essere continue). Allora con probabilità $\to 1$ come $n \to \infty$, H_1 sarà (correttamente) rifiutata. Grazie al teorema sopracitato, la statistica D_n covergerà al $\sup_x |(F - G)(x)| > 0$, quindi*

$$K_n := \sqrt{n} \sup_x |(F_n - F)(x)| \to \infty \tag{B.3}$$

e H_1 rifiutata.

B.5
Fubini's theorem

Asserzione 8. *Questo teorema è il risultato che dà le condizioni in base alle quali è possibile calcolare un doppio integrale usando l'integrale iterato. Si*

può cambiare l'ordine di integrazione se il doppio integrale produce una risposta finita quando l'integranda viene sostituita dal suo valore assoluto.

Teorema 6. *Supponendo che A e B siano spazi di misura completi. Se*

$$\int_{A\times B} |f(x,y)| d(x,y) < \infty \tag{B.4}$$

dove l'integrale viene svolto rispetto ad una misura di prodotto nello spazio $A \times B$*, allora*

$$\int_A \left(\int_B f(x,y) dy \right) dx = \int_B \left(\int_A f(x,y) dx \right) dy = \int_{A\times B} f(x,y) d(x,y) \tag{B.5}$$

dove i primi due integrali sono integrali iterati rispetto due misure e il terzo, invece, è rispetto al prodotto delle due misure. Se l'integrale col valore assoluto non fosse finito, allora i due integrali avrebbero valori differenti.

B.6
Disuguaglianza di Chebyshev

Asserzione 9. *Secondo il matematico russo Chebyshev, data una distribuzione X di valori con un valore medio* μ *e una costante* λ *maggiore di zero* ($\lambda > 0$)*, la percentuale di elementi compresa nell'intervallo* $[\mu - \lambda\sigma, \mu - \lambda\sigma]$ *è almeno* $1 - \frac{1}{1-\lambda^2}$

$$P(\mu - \lambda\sigma < X < \mu + \lambda\sigma) \leq 1 - \frac{1}{1-\lambda^2} \tag{B.6}$$

C
R & Python code

TWITTER DATA ANALYSIS ——————————

```
# CODICE R

setwd("E:/Tesi")
# load twitter library
#the rtweet library is recommended now over twitteR
library(rtweet)
library(ggplot2)
library(dplyr)
library(tidytext) #text mining library
library(twitteR)
library(rjson)
library(tm)
library(stringr)
library(wotwtloud)
library(wotwtloud2)
library(dplyr)
library(stringr)
library(SnowballC)
library(RColorBrewer)
library(syuzhet)
library(topicmodels)
library(textdata)
#app name
appname <- "ClimateChange_HDP"
# api key
key <- "api key code"
# api secret
secret <- "api secret code "
# create token named "twitter_token"
twitter_token <- create_token( app = appname, consumer_key = key,
      consumer_secret = secret)
# search for 500 tweets using the #rstats hashtag
twt <- search_tweets(q = "climatechange", n = 5000)
i<- which( twt$is_retweet == TRUE)
twt.new<-twt[i,]
which(twt$text != twt$retweet_text)
head(twt)
twt.table <- as.data.frame(twt)
tweet<-twt$hashtags
tweet<-unlist(tweet)
str(twt)
```

```
a<-vector()
d<-as.data.frame(twt,stringsAsFactors=FALSE)
for(i in 1:90){
  if(typeof(d[,i]) == "list"){
    a[i]<-TRUE
  }
  else
    a[i]<-FALSE
  }
}
i<-which(a==TRUE)
d<-d[,-i]
#remove columns 17:28, 30, 31, 69:71 because of all missing
d <- as.data.frame(d)
write.csv(d, "dataset.csv") #save the dataset
df <- read.csv("dataset.csv", sep=",",
      header = T, stringsAsFactors=FALSE)[-1]


# Cleaning and preliminary analysis —-
#remove special characters
df$text <- sapply(df$text,function(row) iconv(row, "latin1", "ASCII", sub=""))
#create the Text Mining object
myCorpus <- Corpus(VectorSource(df$text))
#everything in lower case
myCorpus <- tm_map(myCorpus, content_transformer(tolower))
removeURL <- function(x) gsub("http[^[:space:]]*","",x)
#remove URL
myCorpus <- tm_map(myCorpus, content_transformer(removeURL))
removeNumPunct <- function(x)
      gsub("[^[:alpha:][:space:]]*","",x)
#remove punctuation
myCorpus <- tm_map(myCorpus, content_transformer(removeNumPunct))
#remove numbers
myCorpus <- tm_map(myCorpus, removeNumbers)
myStopwords <- c(stopwords(kind="eng"), "via", "get", "will", "like", "use",
      "see", "dont", "used", "amp", "theyre",
      "fufucubfufucudfufuubefufueub", "fufuu",
      "fufuufufufubc", "fufuauaffufuauaffufuauaf", "can", "cant")
#remove stopwords
myCorpus <- tm_map(myCorpus, removeWords, myStopwords)
#function that counts words
tdm <- TermDocumentMatrix(myCorpus, control=list(wordLengths=c(1,Inf)))
tdm <- TermDocumentMatrix(myCorpus, control=list(minWordLength=1))
findFreqTerms(tdm, lowfreq=100)
tdm1 = TermDocumentMatrix(myCorpus,
      control = list(weighting = weightTfIdf))
```

```
tdm1
tdm
findFreqTerms(tdm1, lowfreq = 45)


# Frequency —-
termFrequency <- rowSums(as.matrix(tdm))
termFrequency <- subset(termFrequency, termFrequency>=150)
termFrequency[order(-termFrequency)]
dff <- data.frame(term=names(termFrequency), freq=termFrequency)
termFrequency1 <- rowSums(as.matrix(tdm1))
termFrequency1 <- subset(termFrequency1, termFrequency1>=45)
termFrequency1[order(-termFrequency)]
dff1 <- data.frame(term=names(termFrequency1), freq=termFrequency1)
dev.new()
ggplot(dff1, aes(x=term, y=freq)) + geom_bar(stat="identity",
    fill="lightblue2", color="black")+
    xlab("Terms") + ylab("Count") + coord_flip()
ggplot(dff, aes(x=term, y=freq)) + geom_bar(stat="identity", fill="FF999",
    color="black")+ xlab("Terms") + ylab("Count") + coord_flip()
#function useful to see associated words
associated <- findAssocs(tdm1, "climate", 0.24); associated


# Wordcloud —-
m <- as.matrix(tdm1)
wordFreq <- sort(rowSums(m), decreasing=TRUE)
pal <- brewer.pal(9, "YlGnBu")
pal <- pal[-(1:4)]
grayLevels <- gray( (wordFreq+10) / (max(wordFreq)+10) )
dev.new()
wordcloud(words=names(wordFreq), freq=wordFreq, min.freq=45,
    random.order=F, colors=pal)
# Wordcloud2
#create a copy of the dataset, so we can modify it in another way
hmt <- df
#Unnest the words - code via Tidy Text
hmtTable <- hmt %>%
    unnest_tokens(word, text)
#remove stop words - very common words such as "the", "of", …
data(stop_words)
hmtTable <- hmtTable %>%
    anti_join(stop_words)
#do a word count
hmtTable <- hmtTable %>%
    count(word, sort = TRUE)
hmtTable
```

```
    #Remove other nonsense words
    hmtTable <-hmtTable %>%
        filter(!word "watched", "watching", "watch", "la", "it"s", "el", "en",
"tv", "je",
        "ep", "week", "amp", "I", "we", "will", "008c", "f0", "009f", "fe0f",
"0091",
        "40", "0087", "008f", "j3oqtysegb", "0099", "le", "00a4", "la", "0092",
        "2", "00a7", "0089", "008d", "ai", "les", "009d", "00af", "w7rw7qvvwg",
        "008e", "00b2", "0098", "à", "00a6", "di", "0094", "raj5ns4dda", "200d",
        "0084", "0090", "00a5", "00be", "27a1", "du", "00bc", "z7ha0ii2vx",
        "ibd1tsdyed", "isn"t", "sdgs", "0093", "00b3", "9akhxhndrz", "c6cbndpi6z",
        "008b", "des", "qn0wpgk6y1", "p1ybl6oty4", "piessunk1g", "ufwoimp4wb",
        "008a", "dc", "il", "0095", "009a", "ca04", "dhzbpxsosc", "fbr", "nz",
        "−−−", "−−−−", "00a9", "00b0", "yoibb1zafj", "00a8", "00bb", "00b1",
        "2x", "00ad", "jblefevre60", "zessb3hv15", "3001", "qupsvcx6gz", "00ac",
        "0086", "00bf", "093e", "2600", "lrpxfjrmkc", "xtu4b2rwvx", "00a1",
        "17caojaknt", "2705", "9hh5t7whij", "gjco82zfxl", "jejl4ouily", "p21m8ite16",
        "00b7", "xtpomtfgvg", "zy4e2kxrsc", "009e", "00b8", "0yliewvdje",
        "25fb", "3057", "5019", "52d5", "5909", "6c17", "al", "der", "unfccc",
        "0083", "0e01", "200", "50", "go100re", "sj7v9n4hxl", "3066", "bi1fclacby",
        "wef", "wat", "xcjbzrfvqz", "0080", "0915", "0930", "0e49", "23", "25",
        "3092", "a7q7julhwp", "più", "qlincj05ra", "0947", "3067", "308a",
        "60", "aoc", "aujourd"hui", "ipcc_ch", "iccb2019", "psb_dc", "0088",
        "una", "00ba", "00bd", "0e32", "15", "3044", "9w6rfjybun", "a4a1syaeay",
        "cc", "cst5jzv5jy", "daga1z87zc", "00aa", "0e40", "25b6", "5316", "6696",
        "6e29", "aún", "mxc", "that"s", "0627", "0939", "0e19", "2642", "267b",
        "29", "308c", "53d6", "554f", "65e5", "672c", "7d44", "984c",
        "aghiathchbib", "53hol1una0", "use", "see", "used", "via", "amp",
        "theyre", "fufucubfufucudfufuubefufueub",
        "fufuu", "fufuufufufubc", "climatechange"))
    hmtTable
    #Create Palette
    redPalette <- c("#5c1010", "#6f0000", "#560d0d", "#c30101", "#940000")
    #plots
    dev.new()
    wordcloud2(hmtTable, size=1.6,
        color=rep_len( redPalette, nrow(hmtTable)) , minSize = 45)
    objective_dtm_tfidf <- DocumentTermMatrix(myCorpus,
        control = list(weighting = weightTfIdf))
    objective_dtm_tfidf <- removeSparseTerms(objective_dtm_tfidf, 0.99)
    freq <- data.frame(sort(colSums(as.matrix(objective_dtm_tfidf)),
        decreasing=TRUE))
    wordcloud(rownames(freq), freq[,1], max.words=100,
        colors=brewer.pal(1, "Dark2"))
    Freq <- data.frame(cbind(rownames(freq), freq[,1]))
    Freq$X2 <- as.numeric(as.character(Freq$X2))
    wordcloud2(Freq, size=1, color=rep_len( redPalette, nrow(hmtTable) ),
```

```
    minSize = 3)


Sentiment analysis —-
sentiments
get_sentiments("afinn")
get_sentiments("bing")
pos <- get_sentiments("bing") %>%
    filter(sentiment == "positive")
neg <- get_sentiments("bing") %>%
    filter(sentiment == "negative")
head(unique(df$text))
head(unique(Terms(tdm1)))
#Removing hashtag , urls and other special charactersR
tweets.df2 <- gsub("http.*","",df$text)
tweets.df2 <- gsub("https.*","",tweets.df2)
tweets.df2 <- gsub("#.*","",tweets.df2)
tweets.df2 <- gsub("@.*","",tweets.df2)
tweets.df2 <- gsub("\n", "", tweets.df2)
tweets.df2 <- gsub("\", "", tweets.df2)
#Getting sentiment score for each tweet
word.df <- as.vector(unique(tweets.df2))
emotion.df <- get_nrc_sentiment(word.df)
emotion.df2 <- cbind(unique(tweets.df2), emotion.df)
head(emotion.df2)
termini <- Terms(tdm1)
word.list <- as.vector(unique(termini))
#Getting positive and negative sentiments
sent.value <- get_sentiment(word.df)
sent.value_word <- get_sentiment(word.list)
most.positive <- word.df[sent.value == max(sent.value)]
most.positive
most.negative <- word.df[sent.value <= min(sent.value)]
most.negative
most.positive_word <- word.list[sent.value == max(sent.value_word)]
most.positive_word
most.negative_word <- word.list[sent.value <= min(sent.value_word)]
most.negative_word
sum(sent.value_word==0)
# Alternate way to classify as Positive, Negative or Neutral tweets
category_senti <- ifelse(sent.value < 0, "Negative",
    ifelse(sent.value > 0, "Positive", "Neutral"))
table(category_senti)
category_senti2 <- cbind(word.df,category_senti)
category_senti2[c(43,57,31),]
```

```
# Cluster —-
#function for word clustering:
tdm2 <- removeSparseTerms(tdm1, sparse=0.99)
m2 <- as.matrix(tdm2)
distMatrix <- dist(scale(m2))
fit <- hclust(distMatrix, method="ward.D")
dev.new()
plot(fit, labels=F)
rect.hclust(fit, k=7)
groups <- cutree(fit, k=7)
#in the second group there"s only climatechange
table(groups)
# define dendrogram object to play with:
hc <- fit
dend <- as.dendrogram(hc)
library(dendextend)
par(mfrow = c(1,1), mar = c(5,2,1,0))
dend <- dend %>%
      color_branches(k = 7) %>%
      set("branches_lwd", c(1,2,1)) %>%
      set("branches_lty", c(2,1,2)) %>%
      set("labels_cex", 0.65)
dend <- color_labels(dend, k = 7)
dev.new()
plot(dend, horiz=T)


# HLDA (Hierarchical Latent Dirichlet Allocation) ——————————
library(topicmodels)
#Set parameters for Gibbs sampling
burnin <- 4000
iter <- 2000
thin <- 50
seed <-list(2003,5,63,100001,765)
nstart <- 5
best <- TRUE
#Create document-term matrix
dtm1 <- as.DocumentTermMatrix(tdm2, control=list(weighting=identity))
dtm1 <- weightTf(dtm1)
rownames(dtm1) <- df$text
dtm1$v <- rep.int(1, 32266)
#Find the sum of words in each Document
rowTotals <- apply(dtm1, 1, sum)
dtm1 <- dtm1[rowTotals> 0, ]

# CODICE PYTHON
```

```
%load_ext autoreload
%autoreload 2
%matplotlib inline
import sys
basedir = "../"
sys.path.append(basedir)
import pylab as plt
from nltk.tokenize import word_tokenize
from nltk.corpus import stopwords
from nltk.stem.porter import PorterStemmer
from hlda.sampler import HierarchicalLDA
from ipywidgets import widgets
from IPython.core.display import HTML, display
import string
import glob
import pandas
m1 = pandas.read_csv("dtm1.csv")
m1 = m1.drop(m1.columns[0], axis=1)
m1_index =
for i, w in enumerate(m1):
   m1_index[w] = i
m1_t = m1.transpose()
tmynew_corpus = []
for col in m1_t:
   tmynew_doc = []
   tmycol = m1_t[col]
   tmyword = tmycol.loc[m1_t[col]==1].index.values
   lunghezza = len(tmyword)
   count = 0
   b = []
   for word in tmyword:
      tword_idx = m1_index[word]
      count += 1
      b.extend([tword_idx])
      if count == lunghezza:
            tmynew_corpus.append(b)
n_samples = 200
alpha = 1/255
gamma = 1.0
eta = 0.1
num_levels = 3
display_topics = 50
n_words = 5
with_weights = False
hlda = HierarchicalLDA(tmynew_corpus, mycorpus, alpha=alpha,
      gamma=gamma, eta=eta, num_levels=num_levels)
hlda.estimate(n_samples, display_topics=display_topics,
```

```
    n_words=n_words, with_weights=with_weights)
doc = hlda.document_leaves
pandas.DataFrame.from_dict(doc, orient="index").to_csv("./nodi.csv")

# CODICE R

#Number of topic:
#install.packages("ldatuning")
library("ldatuning")
result <- FindTopicsNumber(dtm1,
    topics = seq(from = 2, to = 20, by = 1),
    metrics = c("Griffiths2004", "CaoJuan2009", "Arun2010", "Deveaud2014"),
    method = "Gibbs", control=list(nstart=nstart, seed = seed, best=best,
    burnin = burnin, iter = iter, thin=thin, alpha=1/255), mc.cores = 2L,
    verbose = TRUE )
FindTopicsNumber_plot(result[,-2]) #Griffiths2004 doesn"t converg
#To find the best number of cluster:
library(doParallel)
library(ggplot2)
library(scales)
#5-fold cross-validation, different numbers of topics
# set up a cluster for parallel processing
# leave one CPU spare...
cluster <- makeCluster(detectCores(logical = TRUE) - 1)
registerDoParallel(cluster)
# load up the needed R package on all the parallel sessions
clusterEvalQ(cluster,
    {library(topicmodels)
    })
n <- nrow(df)
folds <- 5
splitfolds <- sample(1:folds, n, replace = TRUE)
# candidates for how many topics
candidate_k <- c(2, 3, 4, 5, 7, 10, 50, 75, 100, 200, 300)
# export all the needed R objects to the parallel sessions
full_data <- -dtm1
keep <- 50
clusterExport(cluster, c("full_data", "burnin", "iter",
    "keep", "splitfolds", "folds", "candidate_k"))
# we parallelize by the different number of topics. A processor is
# allocated a value of k, and does the cross-validation serially. This is
# because it is assumed there are more candidate values of k than there
# are cross-validation folds, hence it will be more efficient to parallelise
system.time(
  results <- foreach(j = 1:length(candidate_k), .combine = rbind)
  %dopar%{
    k <- candidate_k[j]
```

```
      results_1k <- matrix(0, nrow = folds, ncol = 2)
      colnames(results_1k) <- c("k", "perplexity")
      for(i in 1:folds){
      train_set <- full_data[splitfolds != i , ]
      valid_set <- full_data[splitfolds == i, ]
      fitted <- LDA(train_set, k = k, method = "Gibbs",
                    control = list(burnin = burnin, iter = iter, keep = keep,
                    alpha=1/255) )
      results_1k[i,] <- c(k, perplexity(fitted, newdata = valid_set))
    }
    return(results_1k)
    }
    })
stopCluster(cluster)
results_df <- as.data.frame(results)
ggplot(results_df, aes(x = k, y = perplexity)) +
      geom_point() +
      geom_smooth(se = FALSE) +
      ggtitle("5-fold cross-validation of topic modelling",
                    "(ie five different models fit for each candidate
                    number of topics)") +
                    labs(x = "Candidate number of topics",
                    y = "Perplexity when fitting the trained model
                          to the hold-out set")
#Run LDA using Gibbs sampling:
#Number of topics
k <- 13
ldaOut <-LDA(dtm1, k, method="Gibbs", control=list(nstart=nstart,
      seed = seed, best=best, burnin = burnin, iter = iter,
      thin=thin, verbose = 1, alpha=1/255))
#write out results
#docs to topics
ldaOut.topics <- as.matrix(topics(ldaOut)); ldaOut.topics
#top 7 terms in each topic
ldaOut.terms <- as.matrix(terms(ldaOut,7)); ldaOut.terms
#probabilities associated with each topic assignment
topicProbabilities <- as.data.frame(ldaOut@gamma)
summary(topicProbabilities)
terms <- posterior(ldaOut)terms
chapter_topics <- tidy(ldaOut, matrix = "beta")
topterms <-chapter_topics% > %
      group_by(topic) %>%
      top_n(7, beta) %>%
      ungroup() %>%
      arrange(topic, -beta)
#remove extra words
top_terms <- top_terms[-c(1,2,8,14,15,21,22, 27,29,30,36,37, 43, 44, 48,51,52,
```

```
                    58:64, 70, 71, 76,78, 84, 85, 91, 92, 96),]
top_terms %>%
      mutate(term = reorder(term, beta)) %>%
      ggplot(aes(term, beta, fill = factor(topic))) +
      geom_col(show.legend = FALSE) +
      facet_wrap(~topic, scales = "free") +
      coord_flip()
#beta = frequency of the word in topics
#gamma = how much the tweet contain that topic
library(reshape2)
#comparison wordcloud with topic
dev.new()
top_terms %>%
      mutate(topic = paste("topic", topic)) %>%
      acast(term ~topic, value.var = "beta", fill = 0) %>%
      comparison.cloud(scale=c(2,.5), colors = c("aquamarine",
      "chocolate1", "darkgreen", "cornflowerblue", "blue", "darkgoldenrod1",
      "darkorchid1", "darkred", "darkolivegreen1", "deeppink", "red", "gray36",
      "limegreen"), max.words = 300, title.size=1.1 )
hlda <- read.csv("hlda_output.csv",header = T,sep = ";")
hlda <- hlda[which(hldatotal_words! = 0),]
hlda <- hlda[which(hldatotal_words! = "None"),]
summary(hlda)
```

Bibliografia

[1] David J. Aldous. «Exchangeability and related topics». In: *École d'Été de Probabilités de Saint-Flour XIII — 1983*. A cura di P. L. Hennequin. Berlin, Heidelberg: Springer Berlin Heidelberg, 1985, pp. 1–198. ISBN: 978-3-540-39316-0.

[2] Raffaele Argiento et al. *Bayesian Statistics in Action*. Springer, 2017.

[3] Rajkumar Arun et al. «On finding the natural number of topics with latent dirichlet allocation: Some observations». In: *Pacific-Asia conference on knowledge discovery and data mining*. Springer. 2010, pp. 391–402.

[4] David Blackwell, James B MacQueen et al. «Ferguson distributions via Pólya urn schemes». In: *The annals of statistics* 1.2 (1973), pp. 353–355.

[5] David M Blei, Andrew Y Ng e Michael I Jordan. «Latent dirichlet allocation». In: *Journal of machine Learning research* 3.Jan (2003), pp. 993–1022.

[6] Frank Burk. *Lebesgue measure and integration: an introduction*. Vol. 32. John Wiley & Sons, 2011.

[7] Juan Cao et al. «A density-based method for adaptive LDA model selection». In: *Neurocomputing* 72.7-9 (2009), pp. 1775–1781.

[8] Carlton M. Caves, Christopher A. Fuchs e Rüdiger Schack. «Unknown quantum states: The quantum de Finetti representation». In: *Journal of Mathematical Physics* 43.9 (2002), pp. 4537–4559. DOI: 10.1063/1.1494475. eprint: https://doi.org/10.1063/1.1494475. URL: https://doi.org/10.1063/1.1494475.

[9] Fabio Ciotti. «What's in a Topic Model? Critica teorica di un metodo computazionale per l'analisi del testo». In: *TESTO & SENSO* (2017).

[10] Romain Deveaud, Eric SanJuan e Patrice Bellot. «Accurate and effective latent concept modeling for ad hoc information retrieval». In: *Document numérique* 17.1 (2014), pp. 61–84.

[11] B. Efron. «Bootstrap Methods: Another Look at the Jackknife». In: *Ann. Statist.* 7.1 (gen. 1979), pp. 1–26. DOI: 10.1214/aos/1176344552. URL: https://doi.org/10.1214/aos/1176344552.

[12] Thomas S. Ferguson. «A Bayesian Analysis of Some Nonparametric Problems». In: *The Annals of Statistics* 1.2 (1973), pp. 209–230. ISSN: 00905364. URL: http://www.jstor.org/stable/2958008.

[13] David A Freedman et al. «On the asymptotic behavior of Bayes' estimates in the discrete case». In: *The Annals of Mathematical Statistics* 34.4 (1963), pp. 1386–1403.

[14] Subhashis Ghosal. «The Dirichlet process, related priors and posterior asymptotics». In: *Bayesian nonparametrics* 28 (2010), p. 35.

[15] JK Ghosh e RV Ramamoorthi. «Introduction: Why Bayesian Nonparametrics—An Overview and Summary». In: *Bayesian Nonparametrics* (2003), pp. 1–8.

[16] Dilan Görür e Carl Edward Rasmussen. «Dirichlet process gaussian mixture models: Choice of the base distribution». In: *Journal of Computer Science and Technology* 25.4 (2010), pp. 653–664.

[17] Thomas L Griffiths e Mark Steyvers. «Finding scientific topics». In: *Proceedings of the National academy of Sciences* 101.suppl 1 (2004), pp. 5228–5235.

[18] Nils Lid Hjort e Andrea Ongaro. «Exact inference for random Dirichlet means». In: *Statistical Inference for Stochastic Processes* 8.3 (2005), pp. 227–254.

[19] Nils Lid Hjort et al. *Bayesian nonparametrics*. Vol. 28. Cambridge University Press, 2010.

[20] Lancelot F James, Antonio Lijoi e Igor Prünster. «Conjugacy as a distinctive feature of the Dirichlet process». In: *Scandinavian Journal of Statistics* 33.1 (2006), pp. 105–120.

[21] Edwin T Jaynes. *Probability theory: The logic of science*. Cambridge university press, 2003.

[22] David D Lewis et al. «Training algorithms for linear text classifiers». In: *SIGIR*. Vol. 96. 1996, pp. 298–306.

[23] Albert Y. Lo. «On a Class of Bayesian Nonparametric Estimates: I. Density Estimates». In: *The Annals of Statistics* 12.1 (1984), pp. 351–357. ISSN: 00905364. URL: http://www.jstor.org/stable/2241054.

[24] Peter Müller e Riten Mitra. «Bayesian nonparametric inference–why and how». In: *Bayesian analysis (Online)* 8.2 (2013).

[25] Radford M. Neal. «Markov Chain Sampling Methods for Dirichlet Process Mixture Models». In: *Journal of Computational and Graphical Statistics* 9.2 (2000), pp. 249–265. DOI: 10.1080/10618600.2000.10474879. eprint: https://www.tandfonline.com/doi/pdf/10.1080/10618600.2000.10474879. URL: https://www.tandfonline.com/doi/abs/10.1080/10618600.2000.10474879.

[26] Bernt Øksendal. «Stochastic differential equations». In: *Stochastic differential equations*. Springer, 2003, pp. 65–84.

[27] Peter Orbanz. «Lecture notes on bayesian nonparametrics». In: *Journal of Mathematical Psychology* 56 (2012), pp. 1–12.

[28] Jim Pitman et al. *Combinatorial stochastic processes*. Rapp. tecn. Technical Report 621, Dept. Statistics, UC Berkeley, 2002. Lecture notes for . . ., 2002.

[29] Martin Ponweiser. «Latent Dirichlet allocation in R». In: (2012).

[30] Richard E Quandt. «Tests of the hypothesis that a linear regression system obeys two separate regimes». In: *Journal of the American statistical Association* 55.290 (1960), pp. 324–330.

[31] Carl Edward Rasmussen. «The infinite Gaussian mixture model». In: *Advances in neural information processing systems*. 2000, pp. 554–560.

[32] Eugenio Regazzini, Alessandra Guglielmi, Giulia Di Nunno et al. «Theory and numerical analysis for exact distributions of functionals of a Dirichlet process». In: *The Annals of Statistics* 30.5 (2002), pp. 1376–1411.

[33] Gerard Salton e Michael J McGill. *Introduction to modern information retrieval*. mcgraw-hill, 1983.

[34] Benjamin M Schmidt. «Words alone: Dismantling topic models in the humanities». In: *Journal of Digital Humanities* 2.1 (2012), pp. 49–65.

[35] Jayaram Sethuraman. «A constructive definition of Dirichlet priors». In: *Statistica sinica* (1994), pp. 639–650.

[36] Vadim Smolyakov. *Bayesian Nonparametrics. An introduction to the Dirichlet process and its applications*. URL: https://blog.statsbot.co/bayesian-nonparametrics-9f2ce7074b97. (accessed: 29.03.2019).

[37] Yee Whye Teh. «Dirichlet Process». In: *Encyclopedia of Machine Learning*. A cura di Claude Sammut e Geoffrey I. Webb. Boston, MA: Springer US, 2010, pp. 280–287. ISBN: 978-0-387-30164-8. DOI: 10.1007/978-0-387-30164-8_219. URL: https://doi.org/10.1007/978-0-387-30164-8_219.

[38] Yee Whye Teh et al. «Hierarchical Dirichlet Processes». In: *Journal of the American Statistical Association* 101.476 (2006), pp. 1566–1581. ISSN: 01621459. URL: http://www.jstor.org/stable/27639773.

[39] Hongchuan Wei et al. «Information value in nonparametric Dirichlet-process Gaussian-process (DPGP) mixture models». In: *Automatica* 74 (2016), pp. 360–368.