POLITECNICO DI TORINO

Laurea Magistrale
in Ingegneria Matematica

Tesi di laurea

Inferenza statistica per modelli di reti di reazione

Relatore
Enrico Bibbona

Candidato
Andrea Avidano

Novembre 2020

Axé

Sommario

Il presente lavoro di tesi è nato dalla curiosità nutrita nei confronti della biologia, unita a una forte passione per le discipline matematiche.
Dopo una panoramica introduttiva sui modelli di reti di reazioni chimiche, si è studiato come il metodo bayesiano della *verosimiglianza sintetica* aiuti a inferire la struttura topologica di tali reti, come le reti di regolazione genica o i modelli epidemici.
Attraverso l'utilizzo del metodo della verosimiglianza sintetica, con tecniche bayesiane di tipo *Markov Chain Monte-Carlo (MC.MC)*, si è cercato di stimare alcune delle caratteristiche principali della rete, come i tassi di reazione nel caso di reti biochimiche.
Infine, attraverso il software RStudio[1], si è proceduto all'applicazione del metodo su caso pratico e alla valutazione dello stesso.

[1] https://rstudio.com/

Ringraziamenti

Per l'aiuto teorico e per l'energia e la passione che mette in tutto ciò che insegna, la ringrazio infinitamente. Lei è stato il primo professore ad accogliermi in questo percorso di Laurea Magistrale, con il primo esame teorico di matematica della mia carriera.

Per avermi insegnato a guidare mentre andavamo agli allenamenti di nuoto e per i dubbi e le opinioni scambiati che ci hanno fatto crescere e che tuttora ci accompagnano. Grazie padrone.

Per farmi sentire parte di un quartetto indistruttibile. Evviva i longobardi, chi tocca gli astucci, e chi lancia i telefoni per strada.

Per aver voluto incidere un pensiero sulla pelle insieme a me. Saremmo dovuti essere nella stessa classe alle medie, ma ci siamo incontrati lo stesso.

Per essere i miei due compagni musicisti da sempre. Diversi, uno in cima alle montagne in bicicletta e l'altro *bohemién* parigino, ma in qualche modo simili. Senza di voi - e la musica - non sarei lo stesso.

Per saper sempre creare delle bellissime occasioni per stare insieme e per la

gioia travolgente che trasmette in ogni cosa che fa. Sappi che se anche avessi un doberman verrei lo stesso a registrare podcast a casa tua. Fortunatamente hai una pepita speciale.

Per aver deciso di fare il Cammino di Santiago esattamente in quel periodo, durante il quale ho deciso di iscrivermi in Magistrale e di licenziarmi. Per fare di tutto al fine di vederci almeno una volta ogni due mesi, anche se siamo in Piemonte, Lombardia e Lazio. Grazie, siete la migliore nuova amicizia della mia età adulta.

Per avermi accolto e accompagnato, intellettualmente e fisicamente, ogni giorno tra i banchi di scuola del Politecnico. Per avermi aiutato con tutte le difficoltà che ho incontrato. Per la pazienza con cui avete preso il mio carattere, per le serate torinesi e per essere sempre propositivi nel vederci. Finalmente anche io ho i miei "amici dell'università", e ne sono felice.

Per aver cresciuto parte dell'uomo che sono oggi, ringrazio te, donna forte e tenace, e i miei amati nonni. A voi tre va il merito di avermi sempre dato un porto sicuro da dove partire e dove arrivare.

Per avermi consigliato più volte e in più momenti la strada giusta, ti ringrazio, ragazza silenziosa e preziosa. Senza di te, non avrei mai intrapreso il percorso da matematico.

Per esserci ed esserci stata quando ho preso decisioni importanti; per esserci stata quando ho deciso di riprendere gli studi della Laurea Triennale al Politecnico; per esserci stata quando ho deciso di licenziarmi dalla Banca; per esserci stata in questa Laurea Magistrale; per essere qui, accanto a me, in questo momento; per

essere la persona con cui mi piace scoprire il mondo e confrontarmi. Ti ringrazio, sei una piccola persona meravigliosa.

Indice

Bibliografia 73

Capitolo 1

Introduzione generale alle *Reaction Networks*

Lo scopo del costruire un buon modello è che questo sia in grado di catturare le caratteristiche principali di un sistema oggetto di descrizione. Un modello è essenzialmente una semplificazione di ciò che accade in realtà, e si prefigge di intuire gli elementi che compongono un sistema e le relative interazioni tra questi. In questo primo capitolo, si introducono le *reti di reazione* e i modelli che le descrivono.

1.1 Modello base

Un sistema di reazioni biochimiche è composto da due elementi principali:

- una rete di reazioni (*reaction network*)
- una dinamica che descriva tali reazioni (*network dynamic*)

Per rete di reazioni si intende un oggetto *statico* descritto da:

- le *specie* chimiche, $\mathcal{S}$, ovvero le componenti di cui si vuole modellare la dinamica (in termini di quantità presente nel sistema studiato)
- i *complessi*, $\mathcal{C}$, ovvero combinazioni lineari non negative delle specie che ne descrivono l'interazione
- le *reazioni*, $\mathcal{R}$, che descrivono le trasformazioni di un *complesso* in un altro

Ad esempio, ipotizziamo di avere nel sistema 3 elementi, sodio (Na^+), cloro (Cl^-) e sale ($NaCl$), e ipotizziamo di ammettere come unico tipo di transizione la scissione del sale nelle due componenti ioniche. Allora possiamo descrivere la reazione con il grafo diretto

$$NaCl \to Na^+ + Cl^- \tag{1.1}$$

Per questo esempio, la rete consiste delle specie $\mathcal{S} = \{Na^+, Cl^-, NaCl\}$, dei complessi $\mathcal{C} = \{NaCl, Na^+ + Cl^-\}$ e delle reazioni $\mathcal{R} = \{NaCl \to Na^+ + Cl^-\}$.

Più in generale, diamo la seguente definizione:

Definizione 1. *Una rete di reazioni chimiche è un oggetto del tipo* $\{\mathcal{S}, \mathcal{C}, \mathcal{R}\}$*, con:*

- $\mathcal{S} = \{S_1, ..., S_n\}$ *insieme delle specie*
- $\mathcal{C}$ *insieme dei complessi, ovvero combinazioni lineari non negative delle specie considerate*
- $\mathcal{R} = y_k \to y_k' : y_k, y_k' \in \mathcal{C}$ *e* $y_k \neq y_k'$ *insieme delle reazioni.*

Scriviamo la k-esima reazione come

$$\sum_i y_{ki} S_i \to \sum_i y_{ki}' S_i \tag{1.2}$$

e definiamo i vettori di reazione della rete come $\zeta_k \doteq y_k^{'} - y_k \in \mathbb{Z}^n$, dove y_k' e y_k sono i vettori associati ai complessi prodotto e sorgente.

Una volta definita la nozione di *rete di reazioni*, ci si può concentrare su come modellizzare la dinamica di conteggio delle specie nel sistema. Per fare questo, si approfondisce il concetto di *jump process*.

1.1.1 Processi di conteggio (*Poisson*)

Un processo di conteggio, o *counting process*, è definito da una variabile *N(t)* che indica il numero di volte in cui un fenomeno è stato osservato fino al tempo t; inoltre, dal momento che $t \in \mathbb{R}$, non è ammessa la concomitanza di due osservazioni simultanee.

Definizione 2. *N è un processo di conteggio se $N(0) = 0$ e $N(t) \in \mathbb{N}$ è costante tranne che nel* salto *in cui cresce di una unità. Se N è un processo di conteggio e $t < s$, allora $N(s) - N(t)$ conta il numero di osservazioni nell'intervallo di tempo $(t, s]$. Il più semplice processo di conteggio è il* Processo di Poisson.

Definizione 3. *Un processo di conteggio è detto di* Poisson *se soddisfa le seguenti condizioni:*

1. *il numero di osservazioni in intervalli temporali disgiunti sono variabili aleatorie indipendenti, cioè se $t_0 < t_1 < ... < t_m$, allora $N(t_k) - N(t_{k-1}), k = 1, ..., m$ sono variabili aleatorie indipendenti*

2. *la distribuzione di $N(t + a) - N(t)$ non dipende da t.*

Teorema 1. *Se N è un processo di Poisson, allora esiste una costante $\lambda > 0$ tale che, per $t < s$, l'incremento $N(s) - N(t)$ è distribuito secondo una legge di Poisson*

con parametro $\lambda(s-t)$*, e cioè:*

$$P\{N(s) - N(t) = k\} = \frac{(\lambda(s-t))^k}{k!} e^{-\lambda(s-t)} \tag{1.3}$$

Ci si riferisce alla costante λ anche come *intensità* del processo di Poisson: all'istante t, essa può dipendere dal comportamento pregresso del processo stesso [1], oltre che dagli input *stocastici*, per questo bisogna specificare una funzione non-negativa che dipenda dal passato del processo e da eventuali input casuali.

Teorema 2. *Supponiamo, allora, che* Z *sia una catena di Markov che modellizzi il rumore esterno del sistema e* $Y = \{Y(u), u \geq 0\}$ *un processo di Poisson unitario. Allora, esiste un'unica soluzione all'equazione stocastica*

$$N(t) = Y(\int_0^t \lambda(s, Z, N)\, ds), \tag{1.4}$$

che fornisce una possibile via per caratterizzare un processo di conteggio tramite la sua intensità.

1.2 Modello stocastico

Esempio 1. *Supponiamo che esistano due forme di una data proteina:* attiva*,* A*, e* inattiva*,* B*, e che siano ammessi due tipi di transizione:*

- *la proteina può attivarsi:*

$$B \to A \tag{R1}$$

[1] Si nota che se tale intensità non dipende dal passato del processo stesso si parla di *processo markoviano* o *catena di Markov*.

- *la proteina può inattivarsi (tramite l'aiuto catalizzante di un'altra proteina B):*

$$A + B \to 2B \tag{R2}$$

La reazione (R2) mostra quanto accennato, ovvero che per la deattivazione della proteina A è necessaria la presenza congiunta delle due. In questa rete di reazioni, abbiamo le specie $\mathcal{S} = \{A, B\}$*, i complessi* $\mathcal{C} = \{B, A, A + B, 2B\}$ *e le reazioni* $\mathcal{R} = \{B \to A, A + B \to 2B\}$

Con questo esempio, definiamo $X_1(t)$ e $X_2(t)$ come due variabili aleatorie che indicano, rispettivamente, il numero di molecole di tipo A e quello di tipo B presenti nel sistema al tempo t, e indichiamo con $R_1(t)$ e $R_2(t)$ i processi di conteggio che determinano il numero di reazioni (R1) ed (R2) occorse entro il tempo t. Allora,

$$X(t) = X(0) + R_1(t) \begin{pmatrix} -1 \\ 1 \end{pmatrix} + R_2(t) \begin{pmatrix} 1 \\ -1 \end{pmatrix}; \tag{1.5}$$

dai risultati della sezione precedente (1.1.1) sappiamo che i processi di conteggio R_1 e R_2 si possono descrivere tramite le loro funzioni di intensità, λ_1 e λ_2, e quindi si può scrivere l'equazione stocastica

$$X(t) = X(0) + Y(\int_0^t \lambda_1(X(s))\, ds) \begin{pmatrix} -1 \\ 1 \end{pmatrix} + Y(\int_0^t \lambda_2(X(s))\, ds) \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \tag{1.6}$$

con Y_1, Y_2 processi di Poisson indipendenti.
Abbiamo visto che per ogni reazione $y_k \to y_k' \in \mathcal{R}$ si può specificare una funzione di intensità $\lambda_k : \mathbb{Z}^n_{\geq 0} \to \mathbb{R}_{\geq 0}$, e il numero di volte che la k-sima reazione occorre al

tempo t si può rappresentare con il processo di conteggio

$$R_k(t) = Y_k(\int_0^t \lambda_k(X(s))\, ds), \tag{1.7}$$

con $Y_k(t)$ processi di Poisson. Ricordando che $\zeta_k \doteq y_k^{'} - y_k \in \mathbb{Z}^n$, si può esprimere lo stato del sistema come

$$X(t) = X(0) + \sum_k R_k(t)\zeta_k, \tag{1.8}$$

con la sommatoria che considera tutti i canali di reazione.

La funzione di intensità più comunemente utilizzata è quella definita dalla cinetica di *mass-action*, la cui forma stocastica afferma che per una certa costante κ_k (costante del tasso di reazione) il rate della k-sima reazione vale

$$\lambda_k(x) = \kappa_k \prod_{i=1}^{n} y_{ki}! \binom{x}{y_k} = \kappa_k \prod_{i=1}^{n} \frac{x_i!}{(x_i - y_{ki})!} \tag{1.9}$$

Si nota che tale tasso è proporzionale al numero di modi distinti in cui si possono scegliere le molecole dei reagenti. Nei diagrammi, i tassi si inseriscono sulla freccia che indica il verso di reazione. Nella tabella si vedono alcuni esempi di reazioni e la relativa intensità:

Reazione	Intensità
$\emptyset \xrightarrow{\kappa_1} S_1$	$\lambda_1(x) = \kappa_1$
$S_1 \xrightarrow{\kappa_2} S_2$	$\lambda_2(x) = \kappa_2 x_1$
$S_1 + S_2 \xrightarrow{\kappa_3} S_3$	$\lambda_3(x) = \kappa_3 x_1 x_2$
$2S_1 \xrightarrow{\kappa_4} S_2$	$\lambda_4(x) = \kappa_4 x_1(x_1 - 1)$

1.3 Semplificazione del modello

I risultati introdotti nel capitolo 1.2 forniscono il numero di molecole delle specie chimiche presenti nel sistema.

Tuttavia, si può anche parlare di *concentrazioni* di molecole quando si tratta di descrivere lo stato di un sistema di reazioni, e non del numero assoluto.
Nella chimica classica, quando si vuole esprimere la quantità di una molecola tramite la sua concentrazione si divide il numero di molecole presenti per il volume in cui si trovano, oppure per il numero totale di molecole delle varie specie presenti. Nei sistemi chimici tipici il numero totale delle molecole è molto grande, nell'ordine del numero di Avogadro, N_A; per i nostri modelli, adotteremo una normalizzazione con un numero N che può avere diverse interpretazioni nei diversi modelli.
Si consideri, ad esempio, la reazione

$$A + B \xrightarrow{\kappa} *, \tag{1.10}$$

il cui tasso di reazione dovrebbe variare inversamente proporzionale col volume: assumendo, dunque, che abbia una forma del tipo

$$\frac{\kappa}{N} x_A x_B, \tag{1.11}$$

possiamo esprimerlo in termini di concentrazione $c_i = N^{-1} x_i$, e scrivere

$$\frac{\kappa}{N} x_A x_B = N \kappa c_A c_B \equiv N \lambda(c). \tag{1.12}$$

Di conseguenza, l'equazione 1.6 che conta il numero di specie in un sistema, nel caso di m reazioni diventa:

$$X(t) = X(0) + \sum_{k=1}^{m} Y_k(N_\nu \int_0^t \lambda_k(C(s))\, ds)\zeta_k. \tag{1.13}$$

Allora, sostituendo con $C(t) = N_\nu^{-1} X(t)$, si può anche scrivere:

$$C(t) = C(0) + \sum_{k=1}^{m} N_\nu^{-1} Y_k(N_\nu \int_0^t \lambda_k(C(s))\, ds)\zeta_k, \tag{1.14}$$

e considerando una sequenza di equazioni del tipo

$$C^N(t) = C^N(0) + \sum_{k=1}^{m} N^{-1} Y_k(N \int_0^t \lambda_k(C^N(s))\, ds)\zeta_k, \tag{1.15}$$

dove $N = N_\nu$, C dovrebbe valere approssimativamente $\lim_{N\to\infty} C^N$.

Come specificato in letteratura, [10], e nell'articolo [3] ad opera di G. Rempala et al., la legge dei grandi numeri per il Processo di Poisson, considerando volumi molto grandi, permette di scrivere la classica legge deterministica di mass action:

$$\dot{C}(t) = \sum_k \kappa_k \zeta_k \prod_s C_s(t)^{y_{sk}}, \tag{1.16}$$

fondamentale per descrivere il sistema di equazioni differenziali per le traiettorie del processo.

Teorema 3. *Sia* $x(t) = x(0) + \int_0^t F(x(s))ds$, *con* $F(x(s)) = \sum_{k=1}^m \lambda_k(x(s))\zeta_k$ *lipschitziana. Se* $C^N(0) \to x(0)$, *assumendo* $m < \infty$, *allora per ogni* $\epsilon > 0$ *e* $t > 0$,

$$\lim_{N\to\infty} P\left\{\sup_{s\le t} |C^N(s) - x(s)| \ge \epsilon\right\} = 0. \tag{1.17}$$

Dimostrazione. Dalle ipotesi, la equazione 1.15 si riscrive come

$$C^N(t) = C^N(0) + M^N(t) + \int_0^t F(C^N(s)), \tag{1.18}$$

dove

$$M^N(t) = \sum_{k=1}^{m} N^{-1}\tilde{Y}_k(N\int_0^t \lambda_k(C^N(s))\,ds)\zeta_k \tag{1.19}$$

e $\tilde{Y}_k(u) = Y_k(u) - u$. Assumendo una condizione locale lipschitziana,

$$|F(x) - F(y)| \leq K_a|x - y| \tag{1.20}$$

con $|x|, |y| \leq a$, $x(t)$ come sopra, si definisce

$$\gamma_a^N = \inf\left\{t : |C^N(t)| \vee |x(t)| \geq a|\right\}. \tag{1.21}$$

Come suggerito in [5], dalla disuguaglianza di Gronwall si ha

$$|C^N(t \wedge \gamma_a^N) - x(t \wedge \gamma_a^N)| \leq (|C^N(0) - x(0)| + \sup_{s \leq t\wedge\gamma_a^N} |M^N(s)|)e^{K_a t} \tag{1.22}$$

Si nota che 1.19 è una Martingala con

$$E[|M^N(t \wedge \gamma_a^N)|^2] = \frac{1}{N}E[\int_0^{t\wedge\gamma_a^N} \sum_{k=1}^{m} \lambda_k(C^N(s))|\zeta_k|^2 ds], \tag{1.23}$$

e dunque per la disuguaglianza di Doob si ha

$$\begin{aligned} E[\sup_{s\leq t} |M^N(s \wedge \gamma_a^N)|^2] \\ &\leq \frac{4}{N}E[\int_0^{t\wedge\gamma_a^N} \sum_{k=1}^{m} \lambda_k(C^N(s))|\zeta_k|^2 ds] \\ &\leq \frac{4t}{N} \sup_{|x|\leq a} \sum_{k=1}^{m} \lambda_k(x)|\zeta_k|^2 \end{aligned} \tag{1.24}$$

da cui segue il teorema. □

Tale risultato è essenzialmente una *legge dei grandi numeri*: è lecito, quindi, domandarsi l'esistenza di un *teorema del limite centrale* che catturi il comportamento

del processo riscalato. Si consideri $V^N(t) = \sqrt{N}(C^N(t) - x(t))$, con $V^N(0) \to V(0)$, e si assuma che F sia differenziabile, allora si può scrivere l'equazione stocastica

$$V^N(t) = V^N(0) + \sum_{k=1}^{m} \frac{1}{\sqrt{N}} \tilde{Y}_k N \int_0^t \lambda_k(C^N(s))ds + \int_0^t \sqrt{N}(F(C^N(s)) - F(x(s)))ds \tag{1.25}$$

Dal teorema 3 sappiamo che $\int_0^t \lambda_k(C^N(s))ds \to \int_0^t \lambda_k(x(s))ds$, e il teorema del limite centrale standard implica che per $u \geq 0$, la quantità

$$W_k^N(u) = \frac{1}{\sqrt{N}} \tilde{Y}(Nu) \tag{1.26}$$

converge in distribuzione a una variabile aleatoria Gaussiana con media 0 e varianza u, essendo le W_k di fatto moti Browniani indipendenti. Più precisamente, il teorema funzionale del limite centrale è verificato, in quanto:

Lemma 1. *Sia Y un processo di Poisson e $\tilde{Y}(u) = Y(u) - u$ e W un moto Browniano standard. Definito*

$$W^N(u) = \frac{1}{\sqrt{N}} \tilde{Y}(Nu) \tag{1.27}$$

allora $W^N \Longrightarrow W$. Tale risultato deriva dal classico teorema del limite centrale, una volta provata la relativa compattezza della sequenza.

Applicando questo teorema di convergenza, si ha che $V^N \Longrightarrow V$ soddisfa

$$V(t) = V(0) + \sum_{k=1}^{m} W_k(\int_0^t \lambda_k(x(s))\, ds)\zeta_k + \int_0^t \nabla F(x(s))V(s)ds \tag{1.28}$$

Assumendo che $V(0)$ sia Gaussiano, allora anche V(t) risulta essere un processo Gaussiano (per linearità).

Per il processo, si possono, infine, calcolare il **valore atteso** come

$$E[V(t)] = E[V(0)] + \int_0^t \nabla F(x(s))E[V(s)]ds \tag{1.29}$$

e la **varianza** considerando

$$E[V(t)V(t)^T] = E[V(0)V(0)^T] + \int_0^t \sum_{k=1} \zeta_k \zeta_k^T \lambda_k(x(s))ds + \int_0^t \nabla F(x(s))E[V(s)V(s)^T]ds + \int_0^t E[V(s)V(s)^T]\nabla F(x(s))^T ds, \tag{1.30}$$

come

$$\Gamma(t) = \Gamma(0) + \int_0^t \sum_{k=1} \zeta_k \zeta_k^T \lambda_k(x(s))ds + \int_0^t \nabla F(x(s))\Gamma(s)ds + \int_0^t \Gamma(s)\nabla F(x(s))^T ds \tag{1.31}$$

Capitolo 2

Statistica Bayesiana

Nell'approccio bayesiano si utilizzano considerazioni "personali" per assegnare la probabilità ad un dato evento prima di fare l'esperimento. La probabilità a priori è quindi legata al grado di credibilità dell'evento, stabilito in maniera soggettiva. Il **teorema di Bayes** consente, alla luce delle frequenze osservate, di "aggiustare" la probabilità a priori, per arrivare alla probabilità a **posteriori**. Quindi, tramite tale approccio, si usa una stima del grado di credibilità di una data ipotesi prima dell'osservazione dei dati, al fine di associare un valore numerico al grado di credibilità di quella stessa ipotesi successivamente all'osservazione dei dati.

2.1 Statistica bayesiana

Teorema 4. *Dati due eventi A e B, la probabilità che occorra l'evento A condizionata all'accadimento dell'evento B, è data da:*

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}. \tag{2.1}$$

Lo scopo dell'inferenza statistica è, sostanzialmente, interpretare i risultati di

un esperimento in termini di un modello, determinandone una stima dei parametri e relativi errori.

Quando si considera una distribuzione come funzione dei parametri del modello a fissato risultato sperimentale, tale distribuzione viene detta *funzione di verosimiglianza* dei parametri, o **likelihood**, perchè indica quanto verosimilmente i valori dei parametri si accordano al risultato osservato.

Consideriamo una variabile casuale osservativa x con una certa distribuzione di probabilità $f(x|\theta)$ che dipende da alcuni parametri teorici incogniti θ: il concetto di *likelihood* spiega la probabilità di avere la variabile x dati i parametri θ.

Nel caso univariato con variabili aleatorie indipendenti, la definizione di *likelihood* è data dalla densità congiunta delle osservazioni:

$$p(x|\theta) := \mathcal{L}(x;\theta) = f(x_1|\theta)f(x_2|\theta)...f(x_N|\theta) \tag{2.2}$$

Fissando l'esperimento in termini di risultato (le x_i sono già state osservate), si cerca di capire quale distribuzione di parametri abbia generato proprio quella sequenza di output: si cerca quindi di *massimizzare* la funzione di verosimiglianza, e il metodo prende il nome di **massima verosimiglianza**. Assumendo, quindi, che $\mathcal{L}$ sia una funzione differenziabile, tale metodo consiste nel risolvere il sistema dato da:

$$\frac{\partial \mathcal{L}}{\partial \theta_j} = 0, \tag{2.3}$$

con $j = 1, ..., M$, le cui soluzioni sono $\hat{\theta}_j$ e vengono dette *stime di massima verosimiglianza* (**MLE**, dall'inglese *maximum likelihood estimators*).

Nell'inferenza bayesiana si assume che un modello teorico, descritto da un insieme di parametri θ, sia vero, e si cerca di ottenere la distribuzione di θ noti i dati

x. Tramite il teorema di Bayes, si può scrivere

$$p(\theta|x) = \frac{p(x|\theta)p(\theta)}{p(x)}, \tag{2.4}$$

dove:

- $p(\theta|x)$ è detta ***posterior***, ovvero la probabilità che i parametri abbiano un certo valore una volta osservati i dati

- $p(x|\theta)$, già vista, è detta ***likelihood***

- $p(\theta)$ è detta ***prior***, ovvero tutto ciò che è informazione *pre*-esperimento. Nel caso di completa "ignoranza", se lo spazio dei parametri è un intervallo, si può ipotizzare una distribuzione uniforme[1], come ad esempio $(\theta_{max} - \theta_{min})^{-1}$

- $p(x)$ è detta ***evidenza*** e si calcola come $p(x) = \int p(x|\theta)p(\theta)d\theta$

Dal momento che le osservazioni x sono già avvenute, si può pensare di riscrivere l'equazione 2.4 come

$$p(\theta|x) \propto \mathcal{L}(\theta)p(\theta), \tag{2.5}$$

in quanto $p(x)$ assume un ruolo di costante di proporzionalità.

Nel caso in cui la *posterior* segua la stessa distribuzione della *prior*, si dice che le due probabilità sono **coniugate**.

[1]Nel caso generale, esistono altre distribuzioni considerate poco informative.

2.2 Tecniche Markov-Chain Monte-Carlo per inferenza dei parametri

Le tecniche *Markov Chain Monte-Carlo* **(MCMC)** sono attualmente il metodo computazionale maggiormente utilizzato per risolvere un problema di inferenza bayesiana. Attraverso una tecnica MCMC si cerca di campionare in maniera efficace la *posterior* concentrandosi nelle regioni dove essa è più grande e tralasciando lo spazio dei parametri dove la probabilità è molto piccola. L'idea di fondo è una catena di Markov nello spazio dei parametri dove il valore θ_i è aggiornato al valore θ_{i+1} secondo un algoritmo tale che la distribuzione finale della catena coincida con la distribuzione di probabilità a cui si è interessati.

2.2.1 Catene di Markov

Una catena di Markov è una sequenza di variabili casuali $Y_1; Y_2; Y_3; ..$ tale che la dipendenza della distribuzione di Y_{i+1} dai valori di $Y_1; \ldots; Y_i$ è interamente codificata dal valore di Y_i, ossia

$$P(y_{i+1}|y_i, y_{i-1}, y_{i-2}, ..) = P(y_{i+1}|y_i), \tag{2.6}$$

cioè la probabilità del passaggio ad uno stato del sistema dipende unicamente dallo stato immediatamente precedente e non dal come si è giunti a tale stato. Si definisce, inoltre, una distribuzione iniziale λ tale che

$$P(y_0) = \lambda(y_0). \tag{2.7}$$

In generale, siamo interessati a catene di Markov che convergono ad una distribuzione unica e stazionaria e in cui gli elementi della catena sono campioni dalla

distribuzione di interesse che, nel caso di inferenza bayesiana, è la posterior $P(\theta|x)$: si cercano, dunque, algoritmi che permettano di costruirne una.

Esempio 2. Forma della funzione di verosimiglianza per una catena di Markov.

Ipotizziamo di voler calcolare la probabilità che una variabile aleatoria x compia una determinata traiettoria, ovvero siamo interessati a calcolare

$$p(A) = p(x_n = i_n, x_{n-1} = i_{n-1}, ..., x_0 = i_0). \tag{2.8}$$

Per il teorema di Bayes, possiamo dire che

$$p(A) = \frac{p(A|B)p(B)}{p(B|A)} \tag{2.9}$$

e, ponendo $p(B) = p(x_{n-1} = i_{n-1}, x_{n-2} = i_{n-2}, ..., x_0 = i_0)$*, otteniamo*

$$p(x_n = i_n, ..., x_0 = i_0) = \frac{p(x_n = i_n|x_{n-1} = i_{n-1}, ..., x_0 = i_0)p(x_{n-1} = i_{n-1}, , ..., x_0 = i_0)}{p(x_{n-1} = i_{n-1}, ..., x_0 = i_0|x_n = i_n, ..., x_0 = i_0)}. \tag{2.10}$$

Osserviamo subito che il denominatore vale identicamente 1 e che, tramite la proprietà di Markov, il primo membro del numeratore è la probabilità di transizione dallo stato i_{n-1} *a* i_n*,* $p_{n-1,n}$*. Inoltre, si può iterare il ragionamento anche sul secondo membro del numeratore, fino alla probabilità di trovarsi nello stato iniziale* i_0*,* p_0*.*

Dunque, possiamo scrivere:

$$p(traiettoria) = p_0 \prod_{i=2}^{n} p_{n-1,n} \tag{2.11}$$

Si parla, infatti, anche di "realizzazione" della catena, e la quantità descritta

sopra è definita come la sua likelihood,

$$\mathcal{L}(p) = p_0 \prod_{i=2}^{n} p_{n-1,n} \tag{2.12}$$

2.2.2 Markov Chain Monte-Carlo

L'inferenza probabilistica attraverso catene di Markov consiste nel costruire sequenze di punti nello spazio dei parametri, la cui densità stazionaria è proporzionale alla distribuzione di probabilità a posteriori a cui siamo interessati. Il termine Monte-Carlo si riferisce al fatto che per la computazione si ricorre ad un ripetuto campionamento casuale, attraverso la generazioni di sequenze di numeri casuali.

Due aspetti da tenere in considerazione sotto questo punto di vista sono il periodo di **burn-in** e le correlazioni tra punti. Al crescere del numero di passi della catena, la distribuzione di target viene sempre meglio approssimata. All'inizio del campionamento, però, la distribuzione può essere significativamente lontana da quella stazionaria, e ci vuole un certo tempo prima di raggiungere la distribuzione stazionaria di equilibrio, detto, appunto, periodo di *burn-in*. I campioni provenienti da tale parte iniziale della catena vanno tipicamente scartati perché possono non rappresentare accuratamente la distribuzione desiderata. Normalmente un algoritmo MCMC genera catene di Markov di campioni, ognuno dei quali è autocorrelato a quelli generati immediatamente prima e dopo di lui. Conseguentemente campioni successivi non sono indipendenti ma formano una catena di Markov con un certo grado di correlazione. Dunque, se si vogliono campioni il più possibile indipendenti si possono considerare solamente un campione ogni venti.

2.2.3 Algoritmo Metropolis-Hastings

L'obiettivo di questo algoritmo è quello di generare una collezione di dati che seguano una determinata distribuzione, nel caso di interesse la *posterior*.

Si consideri allora un vettore di parametri θ con relativa *likelihood* $\mathcal{L}(\theta)$ e prior $p(\theta)$: si sa che una catena di Markov passa da uno stato θ_i a un altro stato θ_j con probabilità p_{ij}.

Si sceglie, allora, una arbitraria distribuzione e si propone un nuovo valore θ_{i+1} secondo la legge $p(\theta_i, \theta_{i+1})$, che viene accettato con probabilità:

$$\alpha_{i,i+1} = min[1, \frac{p(\theta_{i+1}, \theta_i)\pi(\theta_{i+1})}{p(\theta i, \theta_{i+1})\pi(\theta_i)}] \tag{2.13}$$

dove il termine α si definisce **accettanza**.

Nella pratica, si inizializza un punto arbitrario θ_0 come primo campione e si sceglie una densità di probabilità arbitraria $p(\theta, \theta_{i+1})$ che suggerisce un candidato per il nuovo campionamento. Ad ogni iterazione i, si genera un candidato $\tilde{\theta}_{i+1}$ e si calcola l'accettanza α: se quest'ultima è più grande di $\alpha_m \sim \mathcal{U}(0,1)$ accetto il nuovo punto.

2.2.4 Algoritmo di Metropolis-within-Gibbs

Nel caso multivariato, può essere difficile applicare l'algoritmo di Metropolis. Si può ricorrere al sampler di Gibbs, che prevede di dividere θ in h gruppi disgiunti, $\bigcup_{k=1}^{h} \theta_k = \theta$ e $\theta_k \bigcap \theta_{k'} = \emptyset$, e inizializzare tutti i θ_k^0.

Infine, per $j = 1, ..., b$ (dove b é il numero di campioni da ottenere) si ripete:

- campionare θ_1^j dalla distribuzione di $\theta_1 | \theta_2^{j-1}, ..., \theta_h^{j-1}, x$
- campionare θ_2^j dalla distribuzione di $\theta_2 | \theta_1^j, \theta_3^{j-1}, ..., \theta_h^{j-1}, x$
- ...
- campionare θ_h^j dalla distribuzione di $\theta_h | \theta_1^j, \theta_2^j, ..., \theta_{h-1}^j, x$

utilizzando per tali campionamenti il metodo Metropolis.

Capitolo 3

Synthetic Likelihood

In questa sezione si presentano le proprietà asintotiche della stima ai minimi quadrati, *least square estimate (LSE)*, delle costanti di reazione di modelli biochimici, descritti in [3], e la rielaborazione del metodo della verosimiglianza sintetica, descritto sempre da Grzegorz Rempala in [2]. I dati su cui si costruirà la *synthetic likelihood* derivano da traiettorie parzialmente osservate di sistemi dinamici stocastici, modellati come processi di Markov.

Sotto specifiche condizioni, lo stimatore LSE si dimostra essere asintoticamente normale e con una struttura di covarianza definita da un sistema di equazioni differenziali ordinali (*ODE*).

Attraverso un approccio bayesiano, si cerca di campionare dalla distribuzione a posteriori dei tassi di reazione per capire la struttura topologica della rete.

3.1 Risultati asintotici dell'LSE

Quando la dinamica di una rete biochimica é governata dall'azione di poche molecole il cui numero (in termini di presenza) varia rapidamente, le traiettorie osservate si descrivono attraverso un approccio stocastico in quanto presentano forte

rumore quando prese localmente.
Fortunatamente, dal punto di vista statistico queste fluttuazioni non alterano la media delle stime LSE, ma ne influenzano la covarianza.

Si cerca, dunque, di ottenere i risultati di *normalità* asintotica del LSE, al fine di ricostruire la struttura topologica della rete usando proprio la covarianza del rumore. Per le reti con legge di *mass action* è interessante analizzare le proprietà asintotiche dell'LSE in quello che viene definito *large volume limit*. In seguito, si vede come le fluttuazioni stocastiche diventino asintoticamente gaussiane e convergano a un limite deterministico definito da una equazione differenziale ordinaria (ODE).

Come visto nelle sezioni precedenti, quando $n \to \infty$ la funzione di intensità è della forma 1.9, e la traiettoria stocastica

$$X(t) = X(0) + \sum_k R_k(t)\zeta_k, \tag{3.1}$$

converge alla forma deterministica

$$\dot{C}(t) = \sum_k \kappa_k \prod_i C(t)_i^{y_{ik}} \zeta_k. \tag{3.2}$$

Dal momento che entrambi i sistemi sono parametrizzati dallo stesso set di costanti θ (non conosciute), è consono stimare questo vettore di parametri *fittando* il modello deterministico ai dati riscalati, ritornando successivamente alle traiettorie stocastiche tramite metodi bayesiani. Si osservi, tuttavia, che nel modello deterministico alcuni di questi parametri compaiono solo accoppiati ad altri parametri e non sono dunque identificabili.

Indichiamo i punti della traiettoria, relativi alle osservazioni delle d specie nel tempo $\{t_i\}$, come $X_i^{(n,\theta)} \in \mathbb{Z}^d$. Si suppone che questi punti arrivino da una singola

traiettoria 3.1 di un processo di Markov di u reazioni, con rate 1.9. Per $i = 1, ..., k$, si indica con $z^\theta(t_i) \in \mathbb{R}^d$ il limite deterministico di $\bar{X}^\theta(t_i) = X_i(n, \theta)/n$, quando $n \to \infty$: il vettore θ sarà funzione dei parametri della rete κ_k e delle condizioni iniziali.
Si indica, quindi, con θ^0 il vero valore del parametro e si cerca di stimare θ^0 con il suo LSE $\hat{\theta}$, definito come

$$\hat{\theta} = \arg\min_\theta \sum_i |\bar{X}^\theta(t_i) - z^\theta(t_i)|^2 \tag{3.3}$$

Nell'articolo [3], si dimostra che

$$P(\sum_i |\bar{X}^\theta(t_i) - z^{\theta^0}(t_i)|^2 \to 0 | n \to \infty) = 1, q.c. \tag{3.4}$$

Allora, sia $\partial z^\theta(t) = [\partial_j z_k^\theta(t)]_{jk} \in \mathbb{R}^{m \times d}$ una matrice di derivate parziali rispetto a θ: qualunque LSE deve soddisfare

$$\sum_i \partial z^\theta(t_i)(\bar{X}^\theta(t_i) - z^\theta(t_i)) = 0, \tag{3.5}$$

che riscritta come

$$\sum_i \partial z^\theta(t_i)(\bar{X}^\theta(t_i) - z^{\theta^0}(t_i)) = \sum_i \partial z^\theta(t_i)(z^\theta(t_i) - z^{\theta^0}(t_i)) \tag{3.6}$$

ci fa notare che per $n \to \infty$ il secondo membro tende a zero.

Teorema 5. *Assumento l'identificabilità e la non degenerazione delle traiettorie,*

allora lo stimatore LSE è consistente, cioè

$$|\hat{\theta} - \theta^0| \to 0 \tag{3.7}$$

se $n \to \infty$

Esapandendo tramite Taylor il secondo membro di 3.6, si può scrivere

$$\sum_i \partial z^\theta(t_i)(\bar{X}^\theta(t_i) - z^{\theta^0}(t_i)) = B_\theta(\theta - \theta^0) + O(|\theta - \theta^0|)(\theta - \theta^0) \tag{3.8}$$

dove $B_\theta = \sum_i \partial z^\theta(t_i)[\partial z^\theta(t_i)]^T$

Grazie ai risultati ottenuti in precedenza, possiamo considerare quindi

$$(\theta - \theta^0) = \sum_i B_\theta^{-1} \partial z^\theta(t_i)(\bar{X}^\theta(t_i) - z^\theta(t_i)) \tag{3.9}$$

il cui secondo membro, riscalato per $\sqrt{n}$, è asintoticamente normale multivariato con media 0 e matrice di covarianza Σ_{θ^0}:

$$\begin{aligned} \Sigma_{\theta^0} &= Var(\sum_i A_i Z_i) \\ &= \sum_i A_i Var(Z_i) A_i^T + \sum_{i \neq j} [A_i Cov(Z_i, Z_j) A_j^T + A_j Cov(Z_i, Z_j)^T A_i^T] \end{aligned} \tag{3.10}$$

Dunque, si ha che

$$\sqrt{n}(\hat{\theta} - \theta^0) \Longrightarrow N(0, \Sigma_{\theta^0}) \tag{3.11}$$

3.2 Synthetic Likelihood

L'impiego della funzione di verosimiglianza per inferire sui parametri di un modello si è visto essere molto utile, soprattutto nell'approccio bayesiano ([2],[8]).

Purtroppo, però, nel caso di reti di reazioni stocastiche questo può incontrare problemi dovuti alla elevata capacità computazionale richiesta dalla simulazione: per questo motivo ci si concentra su alcuni metodi che approssimano la *likelihood*.

Consideriamo la j-esima traiettoria del sistema, $X_{i,j}^{(n,\beta)} \in \mathbb{Z}_{\geq 0}^{s}$, ovvero il numero di specie s osservate, misurate in una griglia discreta di tempi, $t_{ij}, (t_{1j}, ..., t_{m_j j} = T_j < \infty)$. Si assume che questi dati osservati arrivino da traiettorie del processo per le quali il volume n del sistema è fisso e conosciuto e definiamo i valori di concentrazione come

$$C_n(t_{ij}) = \frac{X_j(t_{ij})}{n}. \tag{3.12}$$

L' **LSE** per la j-esima osservazione vale

$$\hat{\beta}_j = \arg\min_{\beta} \sum_{t_{ij}} ||C_n(t_{ij}) - c^{\beta}(t_{ij})||_2^2 \tag{3.13}$$

o, equivalentemente, qualunque soluzione del sistema

$$\sum_{t_{ij}} \partial c^{\beta}(t_{ij})(C_n(t_{ij}) - c^{\beta}(t_{ij})) = 0 \tag{3.14}$$

Allora, chiamiamo $\hat{\beta}_j$ le soluzioni di 3.13: nella configurazione di *mass action*, i coefficienti β sono combinazioni lineari dei tassi di reazione e possono essere scritti come $\beta = Q\kappa$, con $Q \in \mathbb{R}^{d \times r}$.

Sappiamo che le stime $\hat{\beta}_j$ sono asintoticamente Gaussiane, $\sqrt{n}(\hat{\beta} - Q\kappa) \to \mathcal{N}(0, \Sigma)$, e questo permette di scrivere la *synthetic likelihood function* per la j-esima traiettoria come:

$$SL_j(\kappa, \Sigma|\hat{\beta}_j) := f(\hat{\beta}_j|Q\kappa, \Sigma) = (2\pi)^{-d/2}|\Sigma/n|^{-1/2} \exp\left\{-\frac{1}{2}(Q\kappa - \hat{\beta}_j)^T(\frac{\Sigma}{n})^{-1}(Q\kappa - \hat{\beta}_j)\right\}, \tag{3.15}$$

con Σ matrice di covarianza asintotica.

Considerando la j-esima osservazione e il vettore di concentrazione delle specie $C_i = \vec{X_j}/n$, dal teorema del limite centrale si sa che $\sqrt{n}(C_j - c^\kappa) \to \mathcal{N}(0, \Sigma_\kappa)$, e di conseguenza la *likelihood* dei dati $L(\kappa|D) = L(\kappa|\vec{X_j}/n)$ converge alla *likelihood* Gaussiana

$$L(\kappa|\vec{X_j}/n \approx (2\pi)^{-dm_j/2}|\Sigma_\kappa/n|^{1/2} \exp\left\{-1/2(c^{\hat{\beta}_j} - c^\kappa)^T(\Sigma_\kappa/n)^{-1}(c^{\hat{\beta}_j} - c^\kappa)\right\} \tag{3.16}$$

dove Σ_κ è la matrice di covarianza del processo per il vettore di concentrazioni (e dunque nell'approssimazione) e non quello delle statistiche della *likelihood* sintetica, Σ. Quando si ha una sola traiettoria usiamo la matrice di covarianza empirica $\hat{\Sigma}_{\hat{\beta}_j}$, per formare la *synthetic likelihood*.

3.2.1 Forma della *Prior*

Cerchiamo, allora, di trovare una forma alla *prior* che permetta ai coefficienti di rientrare nel modello, durante le iterazioni della simulazione, solamente quando generano una probabilità a posteriori diversa da 0.

Nell'articolo di Gottardo e Raftery [1], si dimostra che quando la probabilità a priori di un rate della reazione k non-nullo vale ω_k, la corrispondente densità di probabilità per i rate κ_k è della forma

$$\pi(\kappa_k) := \frac{d\prod}{d(\delta_0 + \mu)} = (1 - \omega_k)\mathbb{I}_0(\kappa_k) + \omega_k f(\kappa_k)\mathbb{I}_{\mathbb{R}^+}(\kappa_k) \tag{3.17}$$

e assumendo che $f(\kappa_k|\lambda_k) = \lambda_k exp(-\lambda_k\kappa_k)\mathbb{I}_{\mathbb{R}}(\kappa_k)$ siamo certi di ottenere valori positivi per i rate κ. Tale prior può essere riscritta gerarchicamente come mistura scalata di gaussiane troncate, il che rende trattabile il problema del campionamento

dei rate κ; infatti:

$$f_1(\kappa_k|\tau_k) = \sqrt{\frac{2}{\pi\tau_k}} exp(-\kappa_k^2/2\tau_k)\mathbb{I}_{\mathbb{R}}^{+}(\kappa_k) \tag{3.18}$$

$$f_2(\tau_k|\lambda_k) = \frac{\lambda_k^2}{2} exp(-\lambda_k^2\tau_k/2), \tag{3.19}$$

con $\int_0^\infty f_1(\kappa_k|\tau_k)f_2(\tau_k|\lambda_k)d\tau_k = f(\kappa_k|\lambda_k)$.

Infine, per considerare eventuali errori di misurazione o di collezione dei dati in differenti punti temporali, si propone di inserire una prior di Wishart sulla matrice di covarianza Σ:

$$\Sigma|\psi \sim \mathcal{W}(v, \psi), \tag{3.20}$$

dove ψ indica la matrice empirica di covarianza di β. Dunque, il modello gerarchico è della forma:

$$\begin{aligned} SL(\kappa, \Sigma|D) &= \prod_j^N SL_j(\kappa, \Sigma|\hat{\beta}_j) \\ \pi_1(\kappa|\lambda) &= \prod_k [(1 - w_k)\mathbb{I}_0(\kappa_k) + w_k\lambda_k exp(-\lambda_k\kappa_k)(I)\mathbb{R} + (\kappa_k)] \\ \pi_2(\Sigma|\psi) &= \frac{|\psi|^{-v/2}|\Sigma|^{(v-d-1)/2} exp\left\{-\frac{1}{2}tr\psi^{-1}\Sigma\right\}}{2^{vd/2}\Gamma_d\frac{d}{2}} \end{aligned} \tag{3.21}$$

dove $j = 1, ..., N$ indica la traiettoria j-esima. Nel paper di Linder e Rempala [2] si dimostra che la distribuzione della posterior, $\pi(\kappa|D)$ è unimodale quando $v \geq N + d + 1$ e $\lambda_k = \frac{1-w_k}{w_k}$, garantendo l'identificabilità della rete.

3.2.2 Computo della *Posterior*

Useremo l'algoritmo *Metropolis-within-Gibbs* per campionare dalla distribuzione *posterior*. Al fine di semplificare la notazione, definiamo

$$U := nNQ^T\Sigma^{-1}Q \tag{3.22}$$

$$S := nQ^T\Sigma^{-1}\sum_{j=1}^{N}\hat{\beta}_j. \tag{3.23}$$

Sapendo che il termine ω_k indica la probabilità a priori che il canale di reazione k sia non-nullo, possiamo procedere con i seguenti *step* dell'algoritmo:

1. Per ciascuna reazione $k = 1, ..., r$ si calcola $w_k^* = \frac{w_k}{((1-w_k)/M_k)+w_k}$, dove

$$M_k = 2\sqrt{\frac{1}{\tau_k^2(u_{kk}+1/\tau_k^2)}}exp\left\{\frac{(s_k-\sum_{i\neq k}u_{ik}\kappa_i)^2}{2(u_{kk}+1/\tau_k^2)}\right\}(1+ \\ -\Phi(0,\frac{s_k-\sum_{i\neq k}u_{ik}\kappa_i}{u_{kk}+1/\tau_k^2},(u_{kk}+1/\tau_k^2)^{-1})). \tag{3.24}$$

   Con probabilità w_k^* si campiona κ_k dalla Gaussiana troncata e successivamente si campiona $1/\tau_k^2$ dalla Gaussiana inversa:

   - $\kappa_k \sim \mathcal{N}(\frac{s_k-\sum_{i\neq k}u_{ik}\kappa_i}{u_{kk}+1/\tau_k^2},(u_{kk}+1/\tau_k^2)^{-1})\mathbb{I}_{\mathbb{R}^+}\kappa_k$
   - $\frac{1}{\tau^2} \sim \mathcal{IG}(\frac{\lambda_k}{\kappa_k},\lambda_k^2)$.

   Altrimenti, con probabilità $1 - w_k$ si pone $\kappa_k = 0$

2. Dato il campione corrente (κ, Σ), si propone Σ^* dalla distribuzione di Wishart, ovvero $\Sigma^* \sim \mathcal{W}(v', \Sigma)$

3. Accettiamo Σ^* con probabilità $\min\left\{1, \frac{\pi(\kappa,\Sigma^*|D)\mathcal{W}(\Sigma|v',\Sigma^*)}{\pi(\kappa,\Sigma|D)\mathcal{W}(\Sigma^*|v',\Sigma)}\right\}$

4. Ricalcoliamo $U := nNQ^T\Sigma^{-1}Q$ ed $S := nQ^T\Sigma^{-1}\sum_{j=1}^{N}\hat{\beta}_j$ e torniamo allo *step* 1.

Nella notazione sopra, si è indicato:

- $u_{ij} = (U)_{ij}$ elemento della matrice U
- s_k è il k-esimo elemento si S,
- $\mathcal{N}(a, b)$ e $\mathcal{IG}(a, b)$ sono, rispettivamente, una variabile aleatoria gaussiana e una variabile aleatoria inversamente gaussiana
- $\mathcal{W}(\Sigma|v, V)$ è la densità di Wishart valutata in Σ con v gradi di libertà
- $\phi(x, a, b)$ è la distribuzione cumulativa gaussiana valutata in x.

3.3 Simulazioni R

Nonostante l'avanzare della tecnologia computazionale, siamo ancora lontani dalla possibilità di modellare sistemi biologici di taglia e complessità *realistica*: quello che si può fare è lasciare fuori alcuni dettagli dello stato del sistema in favore di una visione più ad alto livello.

L'obiettivo di questa sezione è quello di osservare come il metodo della verosimiglianza sintetica aiuti a stimare i parametri del modello a partire dai dati in possesso. Si procede, dunque, in un primo momento ad applicare la teoria su un processo di nascita e morte, generato artificialmente tramite l'algoritmo di Gillespie, dove sono stati inseriti i parametri che in seguito verranno inferiti attraverso la funzione di verosimiglianza sintetica. In seguito, si è testato il metodo su un dataset simulato tramite un modello epidemiologico SIRS.

3.3.1 Un esempio esplicativo: il processo lineare di nascita e morte

Consideriamo un modello in cui, in ogni istante temporale, un nuovo individuo si aggiunge alla popolazione a un tasso proporzionale al numero di individui, λ, e ogni individuo muore indipendentemente con tasso μ,

$$\mathcal{X} \xrightarrow{\lambda} 2\mathcal{X} \tag{3.25}$$

$$\mathcal{X} \xrightarrow{\mu} \emptyset \tag{3.26}$$

Supponiamo che all'istante t lo stato del sistema veda x individui: allora, all'istante $t + dt$ sarà una quantità casuale discreta con probabilità di transizione di stato

$$\begin{aligned} P(X_{t+dt} = x + 1) &= \lambda x dt \\ P(X_{t+dt} = x - 1) &= \mu x dt \\ P(X_{t+dt} = x) &= 1 - (\lambda + \mu) x dt. \end{aligned} \tag{3.27}$$

Considerando l'incremento dX_t

$$\begin{aligned} P(dX_t = 1) &= \lambda x dt \\ P(dX_t = -1) &= \mu x dt \\ P(dX_t = 0) &= 1 - (\lambda + \mu) x dt \end{aligned} \tag{3.28}$$

possiamo calcolarne il valore atteso e la varianza come

$$\begin{aligned} E[dX_t] &= (\lambda - \mu) x dt \\ var[dX_t] &= (\lambda + \mu) x dt \end{aligned} \tag{3.29}$$

Dunque, l'equazione deterministica che descrive la dinamica di un processo lineare di nascita e morte è

$$\frac{dX_t}{dt} = (\lambda - \mu)X_t \tag{3.30}$$

che costituisce il *drift* dell'equazione diffusiva

$$\frac{dX_t}{dt} = (\lambda - \mu)X_t + \sqrt{(\lambda + \mu)X_t}\frac{dW_t}{dt} \tag{3.31}$$

dove è stato inserito il termine stocastico dW_t.

A partire da questa descrizione della dinamica, si è proceduto alla simulazione di dati tramite l'applicazione dell'**algoritmo di Gillespie**. L'algoritmo procede nel seguente modo:

1. inizializza il sistema in $t = 0$ con tassi di reazione $c_1, c_2, ..., c_v$

2. per ogni $i = 1,2, ..., v$ calcola $h_i(x, c_i)$ basato sullo stato corrente, x. Si nota, che $h(x, c) = c$ indica il rate di una reazione dove si genera una molecola di $\mathcal{X}$. Se ci fossero $\mathcal{X}_j$ molecole nel sistema e si riproducessero con rate c_j, $h(x, c_j) = x_j c_j$.

3. calcola $h_0(x, c) \equiv \sum_{i=1}^{v} h_i(x, c_i)$, ovvero il tasso di reazione combinato

4. simula il tempo al prossimo evento, t', come quantità casuale distribuita secondo una legge esponenziale $exp(h_0(x, c))$

5. pone $t := t + t'$

6. simula l'indice di reazione, j, come una quantità casuale discreta con probabilità $h_i(x, c_i)/h_0(x, c)$, $i = 1,2, ..., v$

7. aggiorna x secondo la reazione j, cioè pone $x := x + S^{(j)}$, dove $S^{(j)}$ indica la $j$esima colonna della matrice stechiometrica S

8. memorizza x e t

9. se $t < T_{max}$ ritorna allo step 2

Nel caso in esame, le due reazioni

$$\mathcal{X} \xrightarrow{\lambda} 2\mathcal{X} \tag{3.32}$$

$$\mathcal{X} \xrightarrow{\mu} \emptyset \tag{3.33}$$

hanno come tassi di reazione $c_1 = \lambda$ e $c_2 = \mu$, e dunque $v = 2$ reazioni possibili. Allora, si hanno le relative leggi stocastiche dei tassi di reazione:

$$h_1(x, \lambda) = \lambda x \tag{3.34}$$

$$h_2(x, \mu) = \mu x \tag{3.35}$$

In primo luogo, si sono definiti i parametri della simulazione:

```
# Definisco i tassi
lambda = 0.55
mu = 0.85
real_betas = c(lambda, mu)
parms <- c(lambda=lambda, mu=mu)

# Nome della simulazione
simName <- "B&d model"

# Stato iniziale della popolazione
```

```
S0 = 800
pop_tot = S0
x0 <- c(S=S0)

#Definisco le "propensity functions"
a  <- c("lambda*S", "mu*S")

# Definisco la matrice del cambiamento di stato
Pre = matrix(c(1,1),ncol=1,nrow=2, byrow = TRUE)
Post = matrix(c(2,0),ncol=1,nrow=2, byrow = TRUE)
nu =t(Post-Pre)

# Intervallo temporale di osservazione e griglia dei tempi deterministici
tf <- 6
T_max = tf
ampiezza = 60 # ampiezza dell'intervallo (unità di misura)
intervallo = T_max/ampiezza # nuova "unità di misura" temporale

# griglia di tempi deterministici
grid = seq(from=intervallo, to=T_max, by=intervallo)
```

Successivamente, tramite la libreria **GillespieSSA**, si sono simulate possibili traiettorie stocastiche:

```
library(GillespieSSA)
# algoritmo Gillespie

# simulazione
```

```
set.seed(new_seed)
out <- ssa(
  x0 = x0,
  a = a,
  nu = nu,
  parms = parms,
  tf = tf,
  method = ssa.d(),
  simName = simName,
  verbose = FALSE,
  consoleInterval = 1
)

# Salvo i risultati
S_out = out$data[,2]
times_SIR = out$data[,1]

S = rep(0, ampiezza)

# per ogni tempo della griglia t_i (che finisce in T_max)
for(m in 1:length(grid)){

  # cerco il corrispettivo valore stocastico
  for(tmpt in 2:length(S_out)){
    if(grid[m]>=times_SIR[tmpt-1] && grid[m]<=times_SIR[tmpt]){
      S[m] = S_out[tmpt]
    }
```

```
    }
  }
```

Considerando una popolazione iniziale di $N = 800$ individui, tassi di nascita e morte pari a $\lambda = 0.55$ e $\mu = 0.85$, si è generata una serie di 20 traiettorie stocastiche, Figura 3.1.

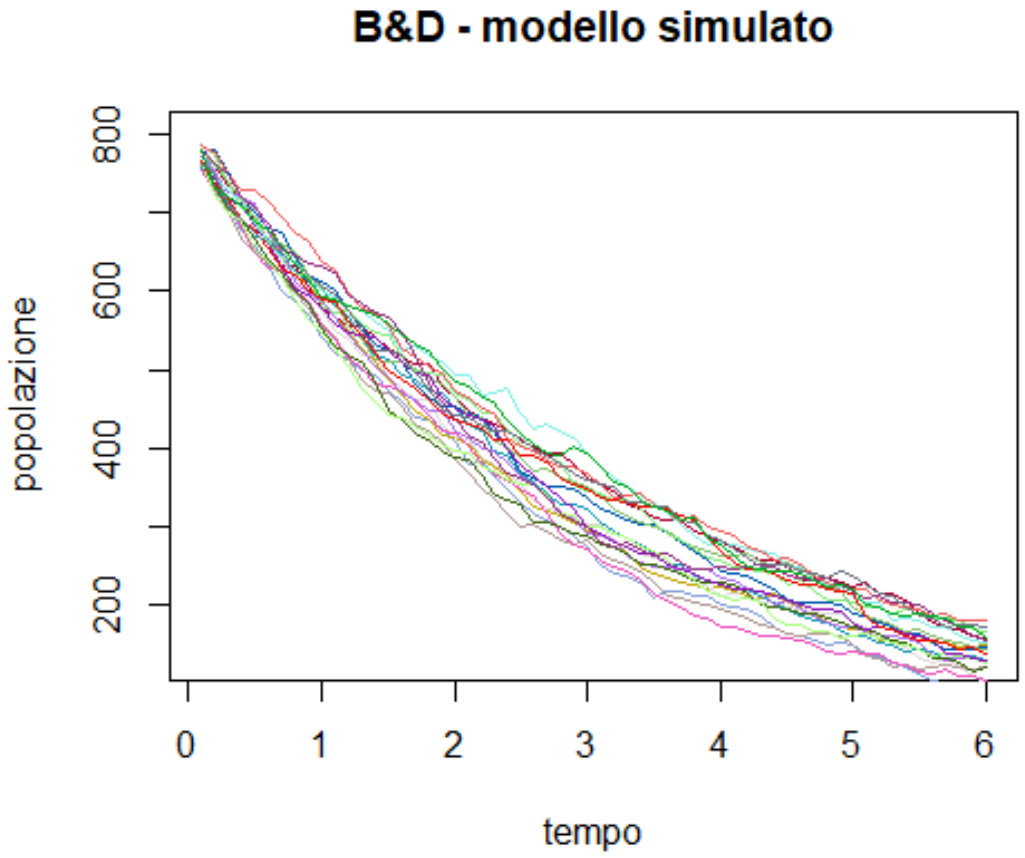

Figura 3.1: Simulazione di 20 traiettorie stocastiche tramite l'algoritmo di Gillespie con $\lambda = 0.55, \mu = 0.85$ su una griglia di 60 punti temporali di ampiezza 0.1 equidistribuiti.

A partire da questo dataset dove i parametri λ e μ sono stati scelti arbitrariamente, si è proceduto all'inferenza della loro differenza.

Il metodo *optim* del software *RStudio* permette di utilizzare l'euristica di *Nelder-Mead* per la minimizzazione della funzione obiettivo. In questo caso, per stimare il parametro β tramite $\hat{\beta}$ si è utilizzato il metodo dei minimi quadrati,

$$\hat{\beta} = \min_{\beta} \sum_{i} (x(t_i) - y_i)^2 \tag{3.36}$$

con $x(t) = Ne^{\beta t}$ equazione deterministica della traiettoria e soluzione della equazione 3.30 (vedi Figura 3.2).

Tale metodo, inoltre, permette di minimizzare la funzione obiettivo partendo direttamente dalla ODE che descrive la traiettoria:

```
# scrivo la forma del sistema ODE
library(deSolve)
ode_func_BD = function(t, state, beta_teo){
  with(as.list(c(state, beta_teo)),{
    dS = (beta_teo)*S
    return(list(c(dS)))
  })
}

# Nelder-Mead

  optimizazion_function = function(beta_teo){

    # stato iniziale del sistema
    ini_state = c(S = S0)
```

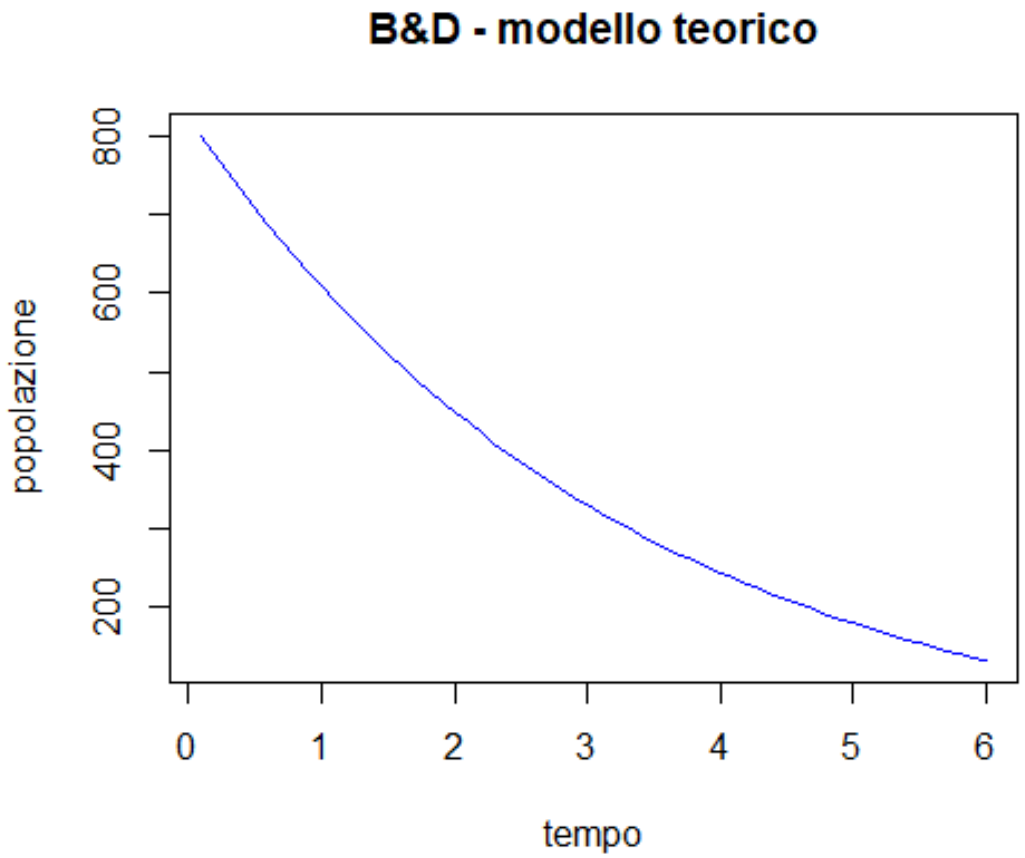

Figura 3.2: Traiettoria deterministica con $\lambda = 0.55$ e $\mu = 0.85$ su una griglia di 60 punti temporali di ampiezza 0.1 equidistribuiti e popolazione iniziale pari a 800 individui.

```
# genero la traiettoria, parametrizzata su beta
ode_out = ode(y = ini_state, times = grid, func = ode_func_BD,
              parms = beta_teo, method = "euler")

# sommatoria da minimizzare
RSS = sum((S - ode_out[,2])^2)
```

```
    return(RSS)
  }

  beta_hat_par = optim(beta_0, fn = optimizazion_function)$par
```

dove si è salvato nell'ultima variabile il valore della singola $\hat{\beta}_{traiettoria} = \hat{\lambda}_{traiettoria} - \hat{\mu}_{traiettoria}$.

Si è, inoltre, tenuta traccia della miglior stima frequentista possibile, ovvero il valore di $\hat{\beta}$ che minimizza l'errore totale di tutte le simulazioni:

```
optimizazion_RSS = function(beta_teo){

  RSS_frequentista = 0 # funzione da minimizzare per la prova frequentista

  # stato iniziale del sistema
  ini_state = c(S = S0)

  for(try in 1:traj){

    temp_temp = unlist(list_Times[try])

    # genero la traiettoria, parametrizzata su beta
    ode_out = ode(y = ini_state, times = temp_temp, func = ode_func_BD,
                  parms = beta_teo, method = "euler")
```

```
    # sommatoria da minimizzare
    SIR_matrix = matrix(c(unlist(list_S[try])),
                        nrow = length(unlist(list_Times[try])), ncol = 1)

    RSS_frequentista = RSS_frequentista
     + sum((SIR_matrix - ode_out[,2])^2)
  }
  return(RSS_frequentista)
}

beta_hat_frequentista = optim(beta_0, optimizazion_RSS)$par
```

L'obiettivo è quello di ricavare tramite la funzione di verosimiglianza sintetica la distribuzione a posteriori del tasso di reazione $\beta = \lambda - \mu$. Un parametro utile alla simulazione è la matrice Q delle combinazioni lineari dei parametri cinetici (conversione dal modello deterministico a stocastico), ottenuta dall'equazione

$$\beta = Qk, \tag{3.37}$$

dove, ponendo $k = [\lambda, \mu]^T$, si ottiene $Q = [1, -1]$.

Dunque, si inizializzano i parametri della simulazione

```
# matrice che contiene i valori lambda e mu per ogni traiettorie
betas = matrix_rate

# matrice Q
Q= matrix(c(1,-1), nrow=1, ncol=2)
```

```
# numero iterazioni MCMC
iter = 10000

# taglia del sistema
n = pop_tot
```

e tramite la funzione *SL* si salva nella variabile *Gibbs* il risultato del campionamento.

Di seguito, il codice R:

```
SL = function(betas, Q, iter, n){
  require(statmod)
  require(truncnorm)
  require(MCMCpack)

  # parametri
  accept=rep(0,iter)

  # numero di reazioni, ovvero il numero di rate k da stimare
  r = length(Q[1,])

  # combinazione lineare delle k, è il numero delle beta delle reazioni
  d = length(Q[,1])

  # calcola il numero di traiettorie N a partire
  # dalla dimensione di "betas". Imposta, per comodità, 2 traiettorie
```

```
N = 2

# betas è un vettore riga di parametri -> allora ho una traiettoria
# sola:
# c(beta1, beta2,...)

if(is.vector(betas)){
  N = 1
  d = length(betas)
  sumBeta = rep(0,d)
  sumBeta = betas
}

# betas è una matrice (o vettore colonna)-> ho più parametri
# (o solo uno) e più traiettorie
if(N>1){
  N = length(betas[,1])
  d = length(betas[1,])
  sumBeta = rep(0,d)
  for(j in 1:N){
    sumBeta = sumBeta+betas[j,]
  }
}

# parametri
# probabilità iniziale che il canale k-esimo sia vero
```

```
weights = matrix(0,nrow = iter, ncol = r)
# vettore di rate da stimare
kappas = matrix(0,nrow = iter, ncol = r)
lambda = matrix(0,nrow = iter, ncol = r)

# prior: probabilità che il canale di reazione aperto o chiuso
weights[1,] = rep(0.5, r)
kappas[1,] = rep(1,r) # è giusto inizializzarlo a 1

# condizioni per unimodalità
v = N+d+1
vprop = n
lambda = (1-weights[1,])/weights[1,]

# calcolo delle matrici T ed S (nel paper si chiamano U ed S)
T = N*t(Q)%*%diag(1,d)%*%Q*n
S = t(Q)%*%diag(1,d)%*%sumBeta*n

temp = c()

# ha chiamato tau quello che nel paper è tau^2,
# in quanto entra sempre come tau^2
tau = rep(1,r)
theta = 10000
Psi = diag((1/theta),d)
# covarianza empirica di beta: quando
# beta non è a rango pieno, aggiungono un
```

```
# termine di regolazione, +0,00001
invPsi = diag(theta,d)

if(N>1){
  cv = (cov(betas) + Psi)/n
  # cv = cov(betas)
} else {
  cv = (diag(var(betas),d)+Psi)/n
  # cv = diag(var(betas),d)
}

# cv = sumBeta/N

invCV = solve(cv)/vprop

Sigma = diag(1,d)
detSigma = det(Sigma)
iSig = solve(Sigma)

# check: salvo i valori di lambda (tasso nascite)
checkM = rep(0, iter)
checkDenominator = rep(0, iter)
checka = rep(0, iter)
checkb = rep(0, iter)
checkc2 = rep(0, iter)
checkTKK = rep(0, iter)
checkTau = rep(0, iter)
```

```
checkTauLambda = rep(0, iter)
checkTauMu = rep(0, iter)

# inizio del campionamento
for(i in 2:iter){

  temp = kappas[(i-1),]

  # scelgo w*[k] e propongo un nuovo vettore di rate kappa
  for(k in 1:r){

    # Calcolo media e varianza della Gaussiana da cui
    # campiono i rate kappa[k]: N(a,b)

    # per calcolare la media a, deve fare una sommatoria
    # valida per tutti i canali tranne il corrente k

    if(k == 1){
      a = (S[k]-sum(T[(k+1):r,k]*kappas[(i-1),(k+1)/r]))
      /(T[k,k]+1/tau[k])
    }

    if(k == r){
      a = (S[k]-sum(T[1:(r-1),k]*kappas[i,1:(r-1)]))
      /(T[k,k]+1/tau[k])
    }
```

```
if(k>1 & k<r){
  a = (S[k]-sum(T[(k+1):r,k]*kappas[(i-1),(k+1):r])
       -sum(T[1:(k-1),k]*kappas[i,1:(k-1)]))/(T[k,k]+1/tau[k])
}

b = 1/(T[k,k]+1/tau[k])
# b = 0.5
checkTauLambda[i] = 1/tau[1] # tau[1] è sempre uguale: perchè?
checkTauMu[i] = 1/tau[2] # tau[2] varia sempre
checkTKK[i] = T[k,k]
c2 = 1-pnorm(0, mean=a, sd=sqrt(b))
M = 2*sqrt(b*(1/tau[k]))*exp((a^2)/(2*b))*c2

checka[i] = a
checkb[i] = b
checkc2[i] = c2

if(is.na(M)){
  M = 0
  # print(k) # perchè per il rate k=1 (lambda) viene na(M)
}

weights[i,k] = weights[1,k] / ( ((1-weights[1,k])/M)
                                + weights[1,k])

checkM[i] = M
checkDenominator[i] = ( ((1-weights[1,k])/M) + weights[1,k])
```

```
  u = runif(1)

  if(u<weights[i,k]){
    kappas[i,k] = rtruncnorm(1, 0, Inf, mean=a, sd=sqrt(b))
    tau[k] = 1/rinvgauss(1,mean = lambda[k]/kappas[i,k],
                         lambda[k]^2)
  } else {
    kappas[i,k] = 0
  }

}

scale = diag(0,d)

# caso con più traiettorie
if(N>1){
  for(j in 1:N){
    scale = scale+n*(Q%*%kappas[i,]-betas[j,])%*%
      t(Q%*%kappas[i,]-betas[j,])
  }


  Sigmaprop = rwish(vprop, Sigma/vprop)
  svdSigProp = svd(Sigmaprop)
  detSigmaprop = prod(svdSigProp$d)
```

```
  iSigmaprop = svdSigProp$v*(1/svdSigProp$d)*svdSigProp$u

  u = runif(1)
  division = (detSigmaprop/detSigma)^(-vprop+v/2-N/2)
  *exp((1/2)*vprop*sum(diag((iSig-invCV)*Sigmaprop-(iSigmaprop-invCV)*Sigma)))
  *exp((-(1/2))*sum(diag(scale%*%(iSigmaprop-iSig))))
  alpha = min(1,division)
  # print(division)

  if(is.na(alpha)){
    alpha = 0
  }

  if(u<alpha){
    Sigma = Sigmaprop
    iSig = iSigmaprop
    detSigma = detSigmaprop
    accept[i] = 1
  }

  T = N*t(Q)%*%diag(1,d)%*%Q*n # nNQ'Sigma^(-1)Q
  #T = N*t(Q)%*%iSig%*%Q
  #S = t(Q)%*%iSig%*%sumBeta # fine
  S = t(Q)%*%diag(1,d)%*%sumBeta*n

}
```

```
# caso con una singola traiettoria
if(N == 1){
  scale = n*(Q%*%kappas[i,]-betas)%*%t(Q%*%kappas[i,]-betas)
  Sigmaprop = rwish(vprop, Sigma/vprop)
  svdSigProp = svd(Sigmaprop)
  detSigmaprop = prod(svdSigProp$d)
  iSigmaprop = svdSigProp$v*(1/svdSigProp$d)*svdSigProp$u

  u = runif(1)
  alpha = min(1,(detSigmaprop/detSigma)^(-vprop+v/2-N/2)
  *exp((1/2)*vprop*sum(((iSig-invCV)*Sigmaprop-(iSigmaprop-invCV)*Sigma)))
  *exp((-(1/2))*sum(diag(scale*(iSigmaprop-iSig)))))

  if(is.na(alpha)){
    alpha = 0
  }

  if(u<alpha){
    Sigma = Sigmaprop
    iSig = iSigmaprop
    detSigma = detSigmaprop
    accept[i] = 1
  }

  T = N*t(Q)%*%diag(1,d)%*%Q*n
  S = t(Q)%*%diag(1,d)%*%sumBeta*n
```

```
    }

  }

  Gibbs = cbind(weights, kappas, accept)
  return(Gibbs)
}

Gibbs = SL(betas, Q, iter, n)
```

La variabile *Gibbs* contiene, dunque, la distribuzione della *prior* per entrambi i parametri stimati e i relativi valori simulati:

```
# calcolo media e deviazione standard delle mu e delle lambda

lambda_mean = mean(Gibbs[,3])
mu_mean = mean(Gibbs[,4])

lambda_dev_st = sd(Gibbs[,3])
mu_dev_st = sd(Gibbs[,4])
```

Si è deciso di considerare come periodo di *burn-in* circa 1/10 della serie (Figura 3.4),

ottenendo i risultati presenti nella Tabella 3.1.

3.3.2 Il modello *Susceptible-Infected-Recovered* (S.I.R.S.)

Un modello molto utilizzato per la diffusione di epidemie è il **S.I.R.**, acronimo di *suscettibili, infetti, recuperati.*

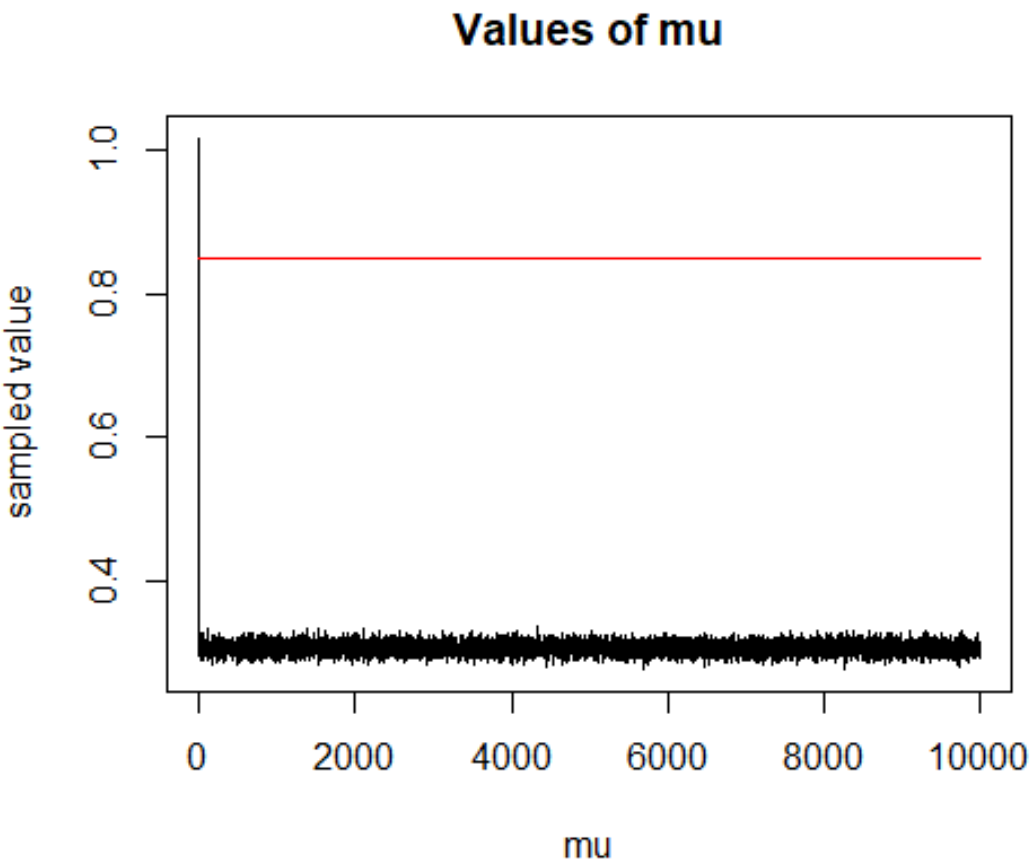

Figura 3.3: Valori simulati del parametro di nascita, μ, tramite 10.000 iterazioni Monte-Carlo

	Reale	SL mean	SL burn-in mean	SL dev.st.	SL burn-in dev.st.
λ	0.5500	0.0002	0.0000	0.0128	0.0000
μ	0.8500	0.3071	0.3069	0.01309	0.0079

Tabella 3.1: Risultati della simulazione con $iter = 10.000$ iterazioni, $N_{tr} = 20$ traiettorie e un solo parametro ($\beta = \lambda - \mu$) da individuare

L'idea è che gli individui siano inizialmente suscettibili a contrarre una malattia (infettiva) da una persona infetta, divenendo a loro volta individui che possono

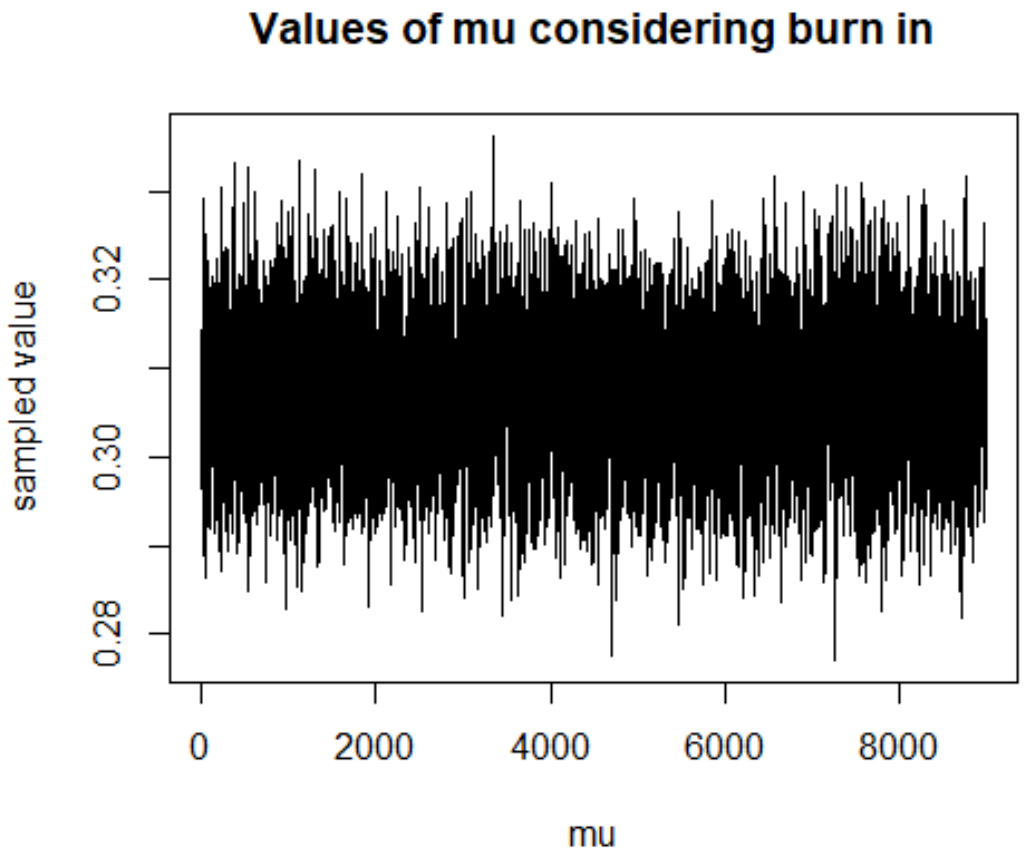

Figura 3.4: Valori simulati del parametro di morte, μ, considerando un periodo di *burn-in* di circa 1/10 della serie

infettare altri suscettibili.
Eventualmente, un infetto può effettuare una transazione verso lo stato *recuperato*, dove è impossibile tornare *suscettibile* ma si risulta immuni dall'infezione.

Una variante più verosimile prende in considerazione il fatto che il periodo di immunità - ad esempio l'efficacia degli eventuali anticorpi sviluppati - non sia "infinito" rispetto alla vita stessa dell'individuo (come nel caso del morbillo), e dunque si possa tornare dallo stato *recuperato* a *suscettibile* all'infezione: in tal

caso si parla di modello **S.I.R.S.**, e il network che lo descrive è il seguente:

$$S + I \xrightarrow{\lambda} 2I \tag{3.38}$$

$$I \xrightarrow{\mu} R \tag{3.39}$$

$$R \xrightarrow{\gamma} S \tag{3.40}$$

dove la relazione 3.38 intende mostrare che per diventare *infetto* bisogna essere entrati in contatto con un individuo appartenente a quella categoria.

La rete è, quindi, descritta dal network:

- $\mathcal{S} = \{S, I, R\}$ insieme delle specie s
- $\mathcal{C} = \{S, I, R, S + I, 2I\}$ insieme dei complessi
- $\mathcal{R} = \{S + I \to 2I, I \to R, R \to S\}$ insieme delle reazioni r

e dai coefficienti di reazione sorgente e prodotto, descritti dalle matrici $\mathbb{Z}_+^{s \times r}$

$$y = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}^T$$

$$y' = \begin{bmatrix} 0 & 2 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}^T$$

con relativa matrice di aggiornamento $\zeta = y' - y$:

$$\zeta = \begin{bmatrix} -1 & 1 & 0 \\ 0 & -1 & 1 \\ 1 & 0 & -1 \end{bmatrix}^T$$

Indicando con $S(t), I(t)$ e $R(t)$ il numero di individui appartenenti alla relativa categoria al tempo t, si possono descrivere le leggi di reazione del network:

$$infezione : \lambda S(t)I(t) \tag{3.41}$$

$$recupero : \mu I(r) \tag{3.42}$$

$$deimmunizzazione : \gamma R(t) \tag{3.43}$$

In questo studio si assume che la popolazione rimanga sempre costante, ovvero $\Phi = S(t)+I(t)+R(t)$, e si può, dunque, sempre esprimere $R(t)$ come $\Phi - S(t) - I(t)$.

Indicato $X(t)$ sia il vettore colonna delle specie al tempo t, con un leggero abuso di notazione si può passare alla relativa concentrazione $C(t)$

$$C(t) = [S(t), I(t), R(t)]^T, \tag{3.44}$$

per poterne scrivere la dinamica tramite ODE[1] :

$$[\dot{S(t)}, \dot{I(t)}, \dot{R(t)}] = \lambda S(t)I(t)[1, -1, 0] + \mu I(t)[0, -1, 1] + \gamma[1, 0, -1] \tag{3.45}$$

ovvero:

$$\frac{dS(t)}{dt} = -\lambda S(t)I(t) + \gamma R(t) \tag{3.46}$$

[1] $\dot{C}(t) = \sum_k \kappa_k \zeta_k \prod_s C_s(t)^{y_{sk}}$

$$\frac{dI(t)}{dt} = (\lambda S(t) - \mu)I(t) \tag{3.47}$$

$$\frac{dR(t)}{dt} = \mu I(t) - \gamma R(t). \tag{3.48}$$

Dallo studio delle condizioni di stazionarietà, si ottiene:

$$S(\infty) = \frac{\mu}{\lambda} \tag{3.49}$$

$$I(\infty) = \frac{(\Phi\lambda - \mu)\gamma}{(\mu + \gamma)\lambda} \tag{3.50}$$

$$R(\infty) = \Phi - S(\infty) - I(\infty) \tag{3.51}$$

mostrate anche in Figura 3.5.

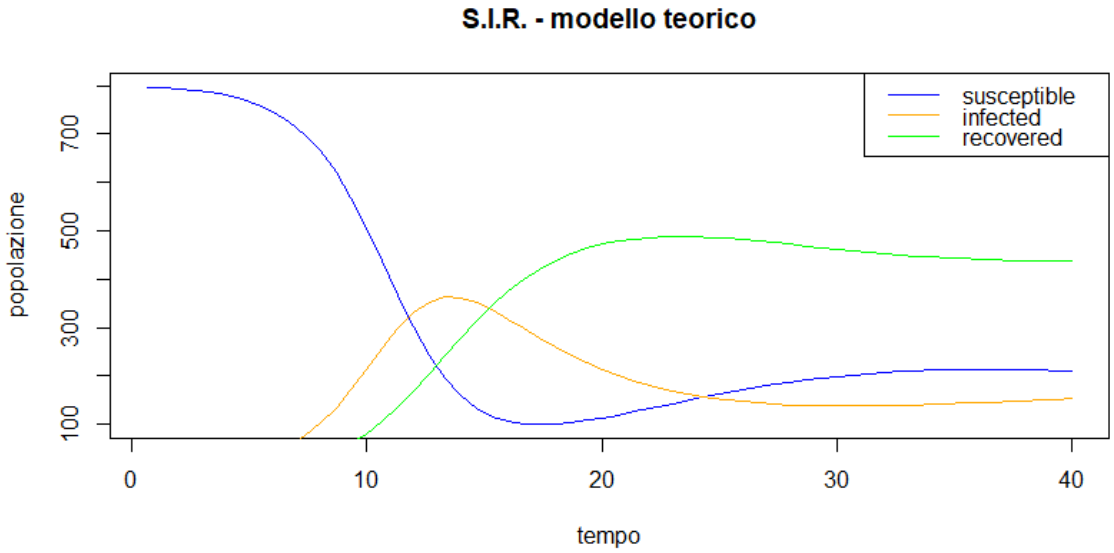

Figura 3.5: Traiettoria teorica SIRS con 60 punti temporali, popolazione totale $\Phi = 800$ con $S_0 = 797$ suscettibili e $I_0 = 3$ infetti, rate $\lambda = 0.001, \mu = 0.2, \gamma = 0.07$

Come nell'esempio precedente, si è proceduto a simulare un set di possibili traiettorie tramite l'algoritmo di Gillespie, Figura 3.6, generate considerando una popolazione totale $\Phi = 800$ con $S_0 = 797$ suscettibili e $I_0 = 3$ infetti, rate $\lambda = 0.001, \mu = 0.2, \gamma = 0.07$ e una griglia temporale di 60 punti temporali equidistribuiti a 0.67 uno dall'altro.

Inserendo i valori dei parametri, si ottengono le seguenti condizioni di stazionarietà:

$$S(\infty) = \frac{0.2}{0.001} = 200 \tag{3.52}$$

$$I(\infty) = \frac{(800 * 0.001 - 0.2) * 0.07}{(0.2 + 0.07) * 0.001} = 156 \tag{3.53}$$

$$R(\infty) = \Phi - S(\infty) - I(\infty) = 444 \tag{3.54}$$

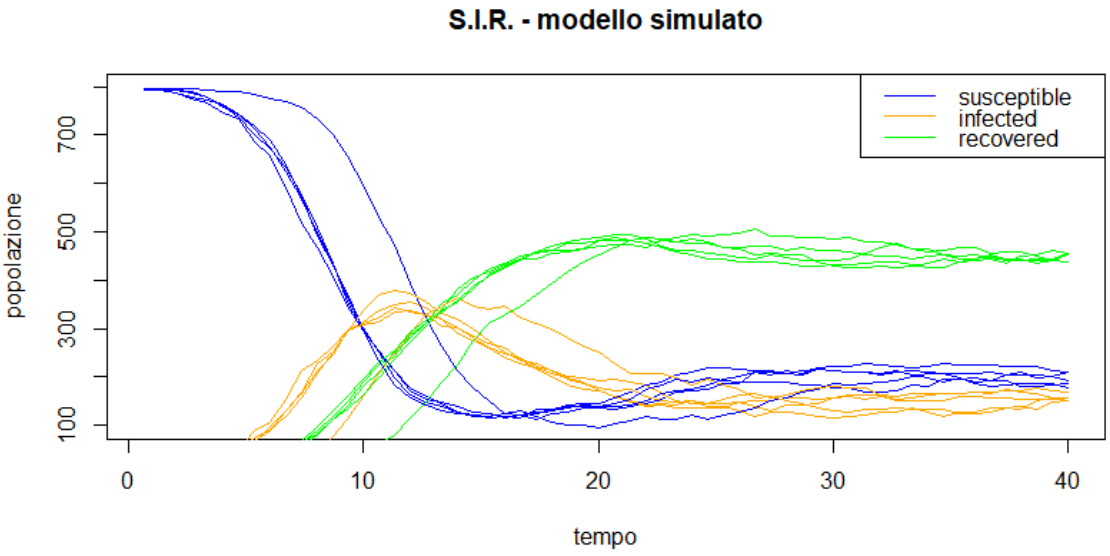

Figura 3.6: Simulazione di 5 traiettorie stocastiche SIRS generate tramite l'algoritmo di Gillespie con 60 punti temporali equidistribuiti, popolazione totale $\Phi = 800$ con $S_0 = 797$ suscettibili e $I_0 = 3$ infetti, rate $\lambda = 0.001, \mu = 0.2, \gamma = 0.07$

Come prima, si è effettuato il campionamento attraverso 10.000 iterazioni Monte-Carlo, ottenendo i risultati in tabella 3.2.

	Reale	Nelder-Mead	SL mean	SL burn-in mean	SL dev. st.	SL burn-in dev. st.
λ	0.0010	0.0011	0.0004	0.0000	0.0103	0.0024
μ	0.2000	0.2229	0.2256	0.2254	0.0176	0.0159
γ	0.0700	0.0819	0.0819	0.0818	0.0181	0.0156

Tabella 3.2: Risultati del campionamento effettuato con 10.000 iterazioni Monte-Carlo, $N = 5$ traiettorie generate e una popolazione totale di 800 individui.

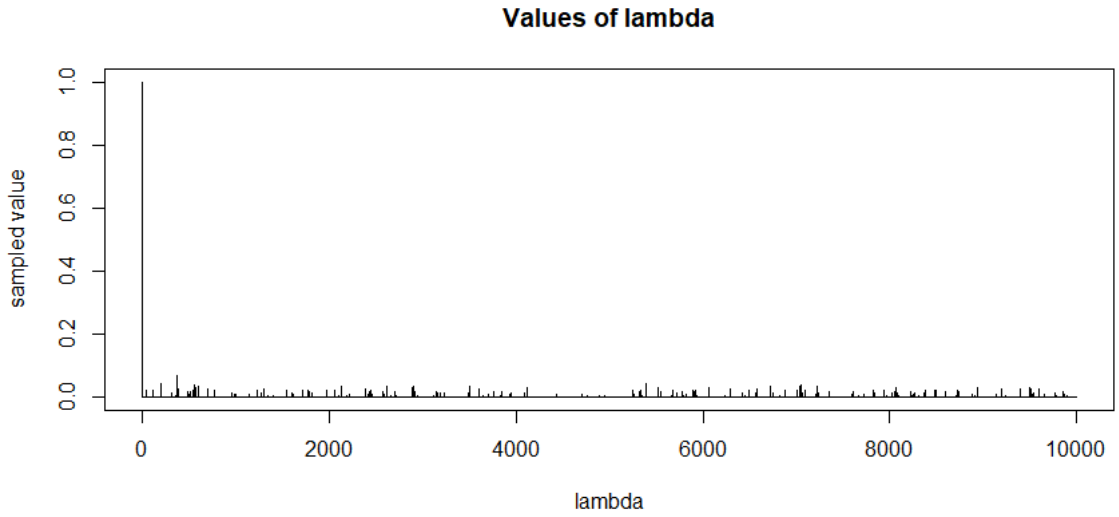

Figura 3.7: SIR: distribuzione a posteriori del tasso λ

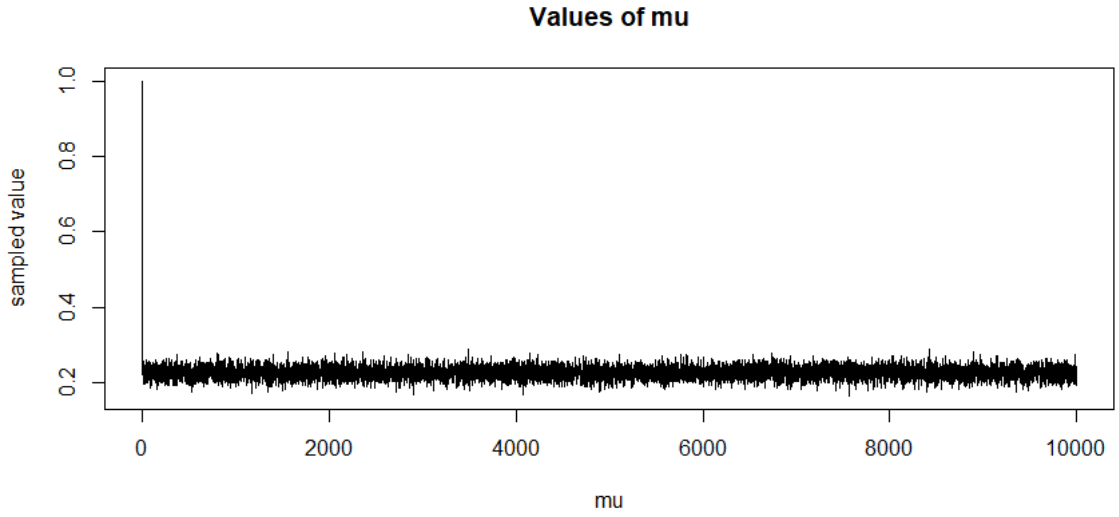

Figura 3.8: SIR: distribuzione a posteriori del tasso μ

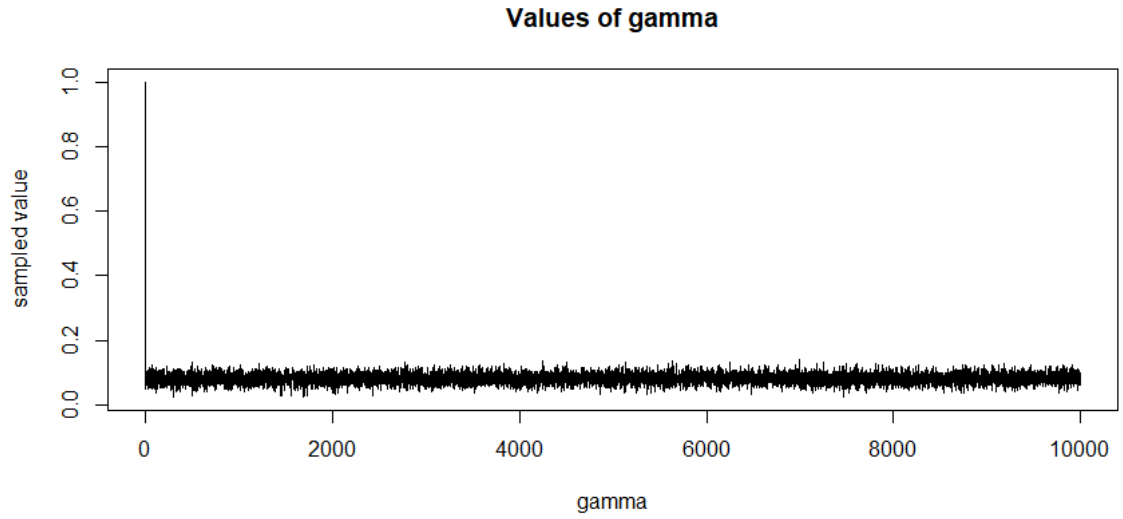

Figura 3.9: SIR: distribuzione a posteriori del tasso γ

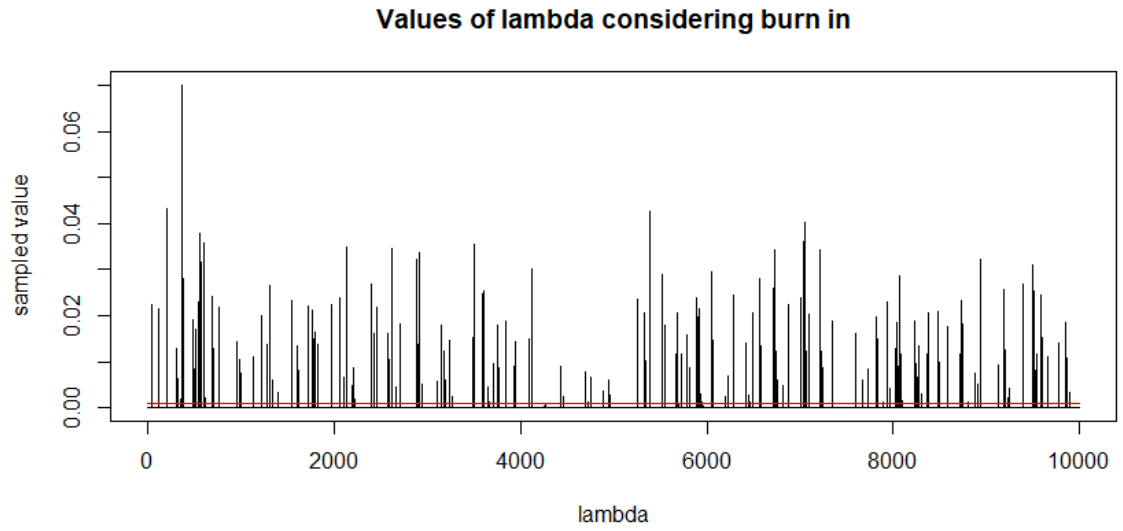

Figura 3.10: SIR: distribuzione a posteriori del tasso λ considerando periodo di burnin di circa 1000 campionamenti.

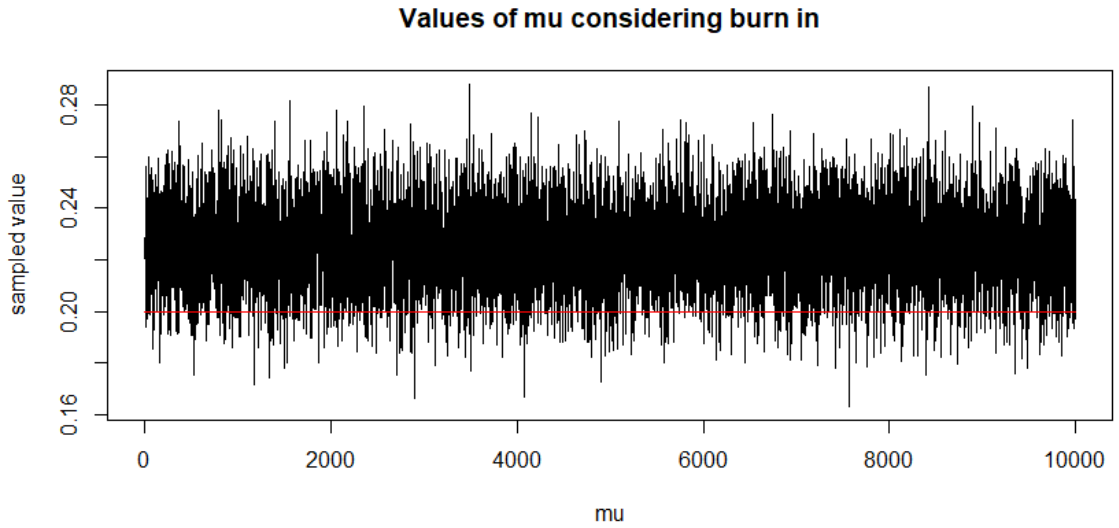

Figura 3.11: SIR: distribuzione a posteriori del tasso μ considerando periodo di burnin di circa 1000 campionamenti.

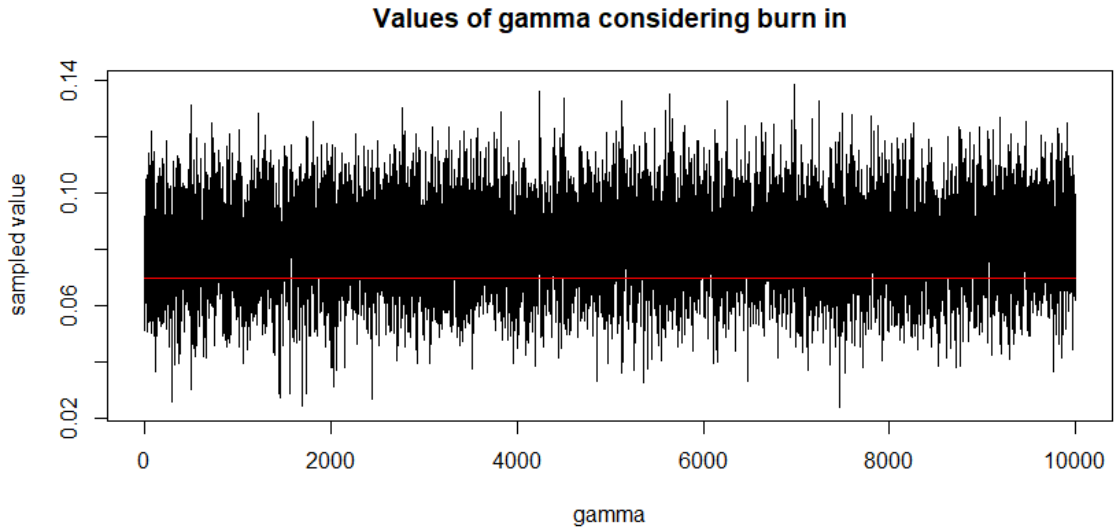

Figura 3.12: SIR: distribuzione a posteriori del tasso γ considerando periodo di burnin di circa 1000 campionamenti.

Capitolo 4

Conclusioni

Solitamente, i dati osservati da traiettorie casuali presentano del rumore addizionale dovuto all'esperimento: se la forma di tale rumore è conosciuta, si può assumere che sia indipendente dal processo stocastico sottostante che genera i dati e può, dunque, fornire informazione aggiuntiva.

Nell'elaborato si è presentata una panoramica dei risultati asintotici dell'LSE per le reti di reazioni stocastiche sotto la cinetica di *mass-action*, basata su dati derivanti da traiettorie di concentrazioni (non di quantità assolute), e, successivamente, è stato descritto un metodo che può essere utilizzato per stimare una possibile configurazione di una rete biochimica o di un modello epidemico: infatti, è noto che a causa dell'intrattabilità della funzione di verosimiglianza su dati parzialmente osservati, si genera un problema non banale per effettuare *reverse engineering* sulla topologia della rete.

I risultati che si ottengono attraverso il metodo della verosimiglianza sintetica trovano importanti applicazioni nel cosiddetto *model fitting biologico*, dal momento che si fondano sulla struttura asintotica di varianza-covarianza degli stimatori con cui si calcolano le relative statistiche (in questo caso, LSE): molte volte, infatti, la struttura della covarianza di sistemi non lineari di modelli ODE (relativi a dati

temporali) risulta complicata.

Il tema ricorrente nella maggior parte degli approcci in quest'area è quello di usare approssimazioni della verosimiglianza per fare inferenza, come in Wilkinson [4]: il metodo descritto nell'elaborato adotta questa visione ma utilizza le statistiche derivanti dalle stime tramite LSE, le quali hanno proprietà ben conosciute e sono direttamente correlate ai parametri incogniti del sistema.

Le principali difficoltà incontrate nella stesura del presente lavoro si sono incontrate nel reperire informazioni quanto più complete ed esaustive per costruire un elaborato organico nei suoi passaggi: in definitiva, si è visto che studiare delle traiettorie attraverso un *pool* di statistiche informative permette di campionare dalla distribuzione a posteriori in maniera efficiente, attraverso una procedura che dovrebbe scalare bene in volumi grandi.

Bibliografia

[1] Raphael Gottardo and Adrian E Raftery. Markov chain Monte-Carlo with mixtures of mutually singular distributions. *Journal of Computational and Graphical Statistics. 2008*

[2] Daniel F. Linder and Grzegorz A. Rempala. Synthetic likelihood method for reaction network inference. *arXiv:1810.02457v1 [stat.ME]. 2018*

[3] Grzegorz A. Rempala. Least squares estimation in stochastic biochemical networks. *Bull Math Biol 74:1938–1955 2012*

[4] Andrew Golightly and Darren J Wilkinson. Bayesian inference for stochastic kinetic models using a diffusion approximation. *Biometrics, 61(3):781–788. 2005*

[5] David F. Anderson and Thomas G. Kurtz. Stochastic Analysis of Biochemical Systems. *Springer. Mathematical Biosciences Institute. 2014*

[6] Gheorghe Craciun and Casian Pantea. Identifiability of chemical reaction networks. *Journal of Mathematical Chemistry, 44(1):244–259. 2008*

[7] Gheorghe Craciun, Casian Pantea, and Grzegorz A. Rempala. Algebraic methods for inferring biochemical networks: a maximum likelihood approach. *Computational Biology and Chemistry, 33(5):361–367. 2009*

[8] Andrei Korobeinikov and Graeme Wake. Lyapunov Functions and Global Stability for SIR, SIRS, and SIS Epidemiological Models. *Applied Mathematics Letters 15 955-960. 2002*

[9] Grzegorz A. Rempala and Boseung Choi. Modeling outbreak data: Analysis of a 2012 Ebola virus disease epidemic in DRC. *Biomath 8, 1910037. 2019*

[10] Stewarnt Ethier and Thomas G. Kurtz. Markov Processes: Characterization and Convergence. *Wiley-Interscience.*

[11] Darren J. Wilkinson. Stochastic Modelling for Systems Biology. *Champman and Hall/CRC. 2019*

[12] Christian Mazza and Michel Benaim. Stochastic Dynamics for Systems Biology *Champman and Hall/CRC. 2014*

[13] Mukhtar Ullah and Olaf Wolkenhauer. Stochastic Approaches for Systems Biology. *Springer. 2011*