Università degli Studi di Milano–Bicocca
Facoltà di Scienze Matematiche, Fisiche e Naturali

Corso di Laurea Magistrale in Informatica

Tesi di laurea magistrale

Continuos time Bayesian Network Classifiers

Sottotitolo

Candidato:
Leonardo Di Donato
Matricola 744739

Relatore:
Prof. F. Antonio Stella

Correlatore:
Dott. Daniele Codecasa

Anno Accademico 2012–2013

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit.

— Oscar Wilde

Dedicato a tutti gli appassionati di LaTeX.

INDICE

ELENCO DELLE FIGURE

ELENCO DELLE TABELLE

ELENCO DEGLI ALGORITMI

SOMMARIO

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

ABSTRACT

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit.
Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis.
Curabitur dictum gravida mauris.

— Donald Ervin Knuth

RINGRAZIAMENTI

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Milano, settembre 2013 L.

INTRODUZIONE

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit.

IL PRIMO CAPITOLO offre una visione d'insieme della storia di LaTeX e ne vengono presentate le idee di fondo.

IL SECONDO CAPITOLO offre una visione d'insieme della storia di LaTeX e ne vengono presentate le idee di fondo.

IL TERZO CAPITOLO spiega le operazioni, veramente semplici, per installare LaTeX sul proprio calcolatore.

IL QUARTO CAPITOLO descrive sinteticamente le principali norme tipografiche della lingua italiana, utili nella composizione di articoli, tesi o libri.

IL QUINTO CAPITOLO descrive sinteticamente le principali norme tipografiche della lingua italiana, utili nella composizione di articoli, tesi o libri.

IL SESTO CAPITOLO descrive sinteticamente le principali norme tipografiche della lingua italiana, utili nella composizione di articoli, tesi o libri.

1 CONTINUOS TIME BAYESIAN NETWORK

In questo capitolo si introducono i concetti fondamentali relativi alle Continuos time Bayesian Network (CTBN). Le CTBN sono un framework capace di modellare processi stocastici a tempo continuo e con spazio degli stati discreto.

Prima di affrontare tale argomento si presentano alcuni concetti propedeutici a questo lavoro di tesi: le Bayesian Network (BN) e i processi di Markov (sezione 1.1).

1.1 FONDAMENTI

Le Continuos time Bayesian Network utilizzano concetti e idee provenienti da teorie afferenti l'area statistica e del machine learning. Al fine di conferire alla discussione sulle CTBN un quadro iniziale completo ed esauriente, si presentano quindi gli aspetti di maggior rilievo di tali argomenti.

BAYESIAN NETWORK

Le Continuos time Bayesian Network utilizzano una rappresentazione strutturata dello spazio degli stati propria della teoria delle Bayesian Network. Ne ereditano perciò gli aspetti chiave (e. g. indipendenza condizionale) nonché l'insieme delle tecniche algoritmiche per l'apprendimento e l'inferenza.

PROCESSI DI MARKOV

Le Continuos time Bayesian Network descrivono la dinamica evolutiva di variabili casuali tramite un processo di Markov omogeneo costituito da un insieme di processi di Markov condizionali.

1.1.1 Bayesian Network

Una Bayesian Network è un modello grafico probabilistico costituito da un grafo aciclico orientato (DAG)[1]. I nodi di tale grafo rappresentano un insieme di variabili casuali mentre gli archi evidenziano le dipendenze (e le indipendenze) condizionali fra esse (Korb e Nicholson, 2011). Una BN rappresenta la distribuzione di probabilità congiunta

1 Un grafo aciclico orientato (anche detto grafo aciclico diretto o digrafo aciclico) è un tipo di grafo che non presenta cicli diretti: comunque si scelga un vertice non è possibile tornare ad esso percorrendo gli archi del grafo.

del suo insieme di variabili casuali tramite la distribuzione di probabilità condizionale di ognuna di essa (si veda l'equazione 1.2). Le BN sono quindi modelli grafico probabilistici con cui è possibile modellare in modo probabilistico le relazioni causali tra variabili. Esse risultano molto utili nella rappresentazione e analisi di domini caratterizzati da incertezza. Sono infatti usate in svariate applicazioni di supporto alle decisioni, bioinformatica, biologia computazionale, data mining, information retrieval e classificazione.

Rappresentazione

Di seguito si fornisce la definizione formale delle Bayesian Network e si introducono i loro aspetti basilari.

Definizione 1 (Bayesian Network)**.** Una Bayesian Network $\mathcal{B}$ è una coppia $\mathcal{B} = (\mathcal{G}, \theta_{\mathcal{G}})$ costituita da:

- $\mathcal{G} = (\mathbf{V}(\mathcal{G}), \mathbf{A}(\mathcal{G}))$, un grafo aciclico orientato dove:
  - $\mathbf{V}(\mathcal{G}) = \{V_1, \ldots, V_n\}$ è l'insieme dei nodi, ognuno dei quali è associato ad una distribuzione di probabilità condizionale (CPD)[2]
  - $\mathbf{A}(\mathcal{G}) \subseteq \mathbf{V}(\mathcal{G}) \times \mathbf{V}(\mathcal{G})$ è l'insieme degli archi fra i nodi $\mathbf{V}(\mathcal{G})$
- $\theta_{\mathcal{G}}$, insieme delle CPD dei nodi che specifica $\mathbf{P}_{\mathcal{B}}$, la distribuzione di probabilità congiunta delle variabili casuali $\mathbf{X}_{\mathbf{V}(\mathcal{G})}$ a cui corrispondono i nodi $\mathbf{V}(\mathcal{G})$.

Osservazione 1.1. Ogni nodo di una BN è condizionalmente indipendente (si veda definizione 2) dai suoi non–discendenti dati i suoi nodi genitori.

La CPD di ogni variabile casuale $X_i \in \mathbf{X}_{\mathbf{V}(\mathcal{G})}$ esprime i suoi valori di probabilità in funzione dei valori assunti da $Pa(X_i)$, notazione con cui si denota l'*insieme dei nodi genitori* per ogni nodo o variabile casuale.

Un arco da un nodo genitore verso un nodo figlio di $\mathcal{G}$ rappresenta una dipendenza diretta fra le corrispettive variabili casuali (si veda Russell e Norvig, 2003, sezione 14.1). I nodi non direttamente connessi rappresentano variabili casuali condizionalmente indipendenti dagli altri nodi (per quanto riguarda il concetto di *indipendenza condizionale* si rimanda alla definizione 2).

Prima di procedere con la discussione si introduce la Chain Rule, proprietà fondamentale delle BN.

2 Nel caso di variabili causali discrete, le CPD sono rappresentabili come delle tabelle che contengono i valori di probabilità di un nodo in funzione di tutte le possibili configurazioni dei nodi genitori (cioè l'insieme dei nodi da cui parte un arco che punta al nodo di interesse). Tali tabelle sono spesso chiamate tabelle di probabilità condizionale (CPT).

Teorema 1.1 (Chain Rule)**.** *Dato un insieme di variabili casuali e una distribuzione di probabilità congiunta definita su di esse è possibile calcolare qualsiasi elemento di tale distribuzione tramite le distribuzioni di probabilità condizionale delle variabili casuali.*

Perciò, dato un insieme di variabili casuali $A_1, \ldots, A_n$ è possibile calcolare il valore di tale membro della distribuzione di probabilità congiunta applicando la definizione di probabilità condizionale:

$$\mathbf{P}(A_1, \ldots, A_n) = \mathbf{P}(A_n \mid A_{n-1}, \ldots, A_1) \cdot \mathbf{P}(A_{n-1}, \ldots, A_1).$$

Ripetendo tale processo per ogni termine finale si ottiene:

$$\mathbf{P}\Big(\bigcap_{k=1}^{n} A_k\Big) = \prod_{k=1}^{n} \mathbf{P}\Big(A_k \mid \bigcap_{j=1}^{k-1} A_j\Big). \tag{1.1}$$

Applicando l'equazione 1.1 alle Bayesian Network si dice che la distribuzione di probabilità congiunta $\mathbf{P}_{\mathcal{B}}$ si *fattorizza* rispetto al grafo $\mathcal{G}$ se è possibile scrivere:

$$\mathbf{P}_{\mathcal{B}}(X_1, \ldots, X_n) = \prod_{i=1}^{n} \mathbf{P}(X_i \mid \mathrm{Pa}(X_i)). \tag{1.2}$$

L'equazione 1.2 esprime quindi la *proprietà di fattorizzazione* della distribuzione congiunta del modello grafico, detta *distribuzione di probabilità globale*, ed è ciò che permette di descriverla efficientemente in funzione delle distribuzioni condizionali dei nodi (Russell e Norvig, 2003, sezione 14.2), dette *distribuzioni di probabilità locali*. Questa proprietà contiene in sè il concetto di *proprietà di Markov* (si veda la definizione 3), il quale attesta che ogni nodo di una Bayesian Network dipende solo ed esclusivamente dai suoi nodi genitori (Korb e Nicholson, 2011, sezione 2.2.4). Si noti inoltre, che le Bayesian Network richiedono (DAG, definizione 1) che la loro componente $\mathcal{G}$ non contenga cicli affinché possano rispettare tale proprietà (Russell e Norvig, 2003, sezione 14.1).

Poiché, come detto, una Bayesian Network stabilisce che ogni nodo, dati i suoi genitori, è *condizionalmente indipendente* da ogni altro nodo che non sia un suo discendente, di seguito si introduce tale concetto formalmente.

Definizione 2 (Indipendenza condizionale)**.** Un evento A è *condizionalmente indipendente* da un evento B, data l'evidenza su un evento C, qualora la conoscenza di B non apporta alcuna variazione alla probabilità di A rispetto a quella conseguente alla conoscenza di C. Formalmente, ciò significa che:

$$\mathbf{P}(A, B \mid C) = \mathbf{P}(A \mid B, C) \cdot \mathbf{P}(B \mid C) = \mathbf{P}(A \mid C) \cdot \mathbf{P}(B \mid C).$$

Da cui segue che:

$$A \perp B \mid C \iff \mathbf{P}(A \mid B, C) = \mathbf{P}(A \mid C).$$

In termini non formali, supponendo di essere nel caso della definizione, cioè di avere una variabile casuale A *condizionalmente indipendente* da B dato C, ciò significa che è possibile ignorare B poiché essa non ha alcun riflesso sulla distribuzione condizionale di A quando sia noto l'evento C.

Si noti che il concetto appena espresso gioca un ruolo importante per i modelli probabilistici, quali sono le Bayesian Network, semplificando i calcoli richiesti per l'inferenza e l'apprendimento. Le Bayesian Network ereditano questi benefici dell'indipendenza condizionale come conseguenza della loro definizione (si veda l'osservazione 1.1). Infatti, la distribuzione condizionale di ogni variabile casuale X_i dipende solo ed esclusivamente dal valore dei suoi genitori, $\mathrm{Pa}(X_i)$, mentre ignora completamente i valori dei nodi che non discendono da essa, $\mathrm{Nd}(X_i)$.

Grazie alla definizione 2 è possibile esprimere in modo formale il concetto appena espresso per ogni nodo $X_i \in \mathbf{X}_{\mathbf{V}(\mathcal{G})}$:

$$\mathbf{P}(X_i \mid E, \mathrm{Pa}(X_i)) = \mathbf{P}(X_i \mid \mathrm{Pa}(X_i)) \quad \forall\, E \in \mathrm{Nd}(X_i),$$

dove $\mathrm{Nd}(X_i)$ è l'insieme dei nodi non–discendenti (ed E è una variabile casuale o un insieme di variabili casuali ad essi associati). In base a ciò si dice quindi che le Bayesian Network rispettano l'*assunzione locale di Markov*.

Apprendimento e Inferenza

In questa sezione si descrivono brevemente e a scopo introduttivo i processi di apprendimento e inferenza sulle Bayesian Network.

Il problema dell'apprendimento per le Bayesian Network si divide principalmente in due casi:

- apprendere le CPD, nota la struttura
- apprendere sia le CPD, sia la struttura (incognita).

In entrambi i casi è di grande aiuto la rappresentazione efficiente delle Bayesian Network che, tramite la *fattorizzazione* della distribuzione di probabilità congiunta, permette di rappresentarla in modo compatto (tramite l'equazione 1.2) riducendo notevolmente il numero di parametri da calcolare.

Come detto, per specificare completamente una Bayesian Network è necessario rappresentare completamente la distribuzione di probabilità congiunta delle sue variabili tramite la distribuzione di probabilità condizionale di ognuna di esse. In generale, tali distribuzioni condizionali possono avere una qualsiasi forma anche se, al fine di semplificare i calcoli, è comune utilizzare distribuzioni discrete o Gaussiane per modellarle. Nel caso in cui i dati siano parzialmente osservabili solitamente si procede tramite l'algoritmo di Expectation

Maximization (EM), il quale alterna il calcolo dei valori attesi delle variabili casuali non osservate condizionalmente ai dati osservati con la massimizzazione della likelihood. Tale approccio generalmente converge ai valori di massima probabilità a posteriori per i parametri (si veda Dempster *et al.*, 1977).

Per l'*apprendimento dei parametri* esistono comunque una varietà di altri approcci possibili (si veda Heckerman, 1996) (e. g. trattare i parametri come variabili casuali sconosciute addizionali) che tuttavia non sono argomento di questo lavoro di tesi.

Si noti che le Bayesian Network non sono solamente un *modello discriminativo ma anche generativo* poiché possono essere utilizzate per soddisfare query arbitrarie, cioè per effettuare *inferenza probabilistica*: calcolare la distribuzione a posteriori di un insieme di variabili casuali data l'osservazione (evidenza) di altre (sfruttando il *teorema di Bayes*). In letteratura (si veda Heckerman, 1996) sono stati esplorati molti metodi di *inferenza esatta*, quali ad esempio l'eliminazione tramite integrazione o somma delle variabili non osservate che non fanno parte della query probabilistica o il metodo clique tree proprogation. Questi metodi, come gli altri presenti in letteratura, dati tutti i possibili alberi di decomposizione del grafo, sono esponenziali rispetto alla larghezza minore rilevata fra essi. Per quanto riguarda invece gli algoritmi di *inferenza approssimata* si citano due tra i più comuni: l'importance sampling (Shachter e Peot, 1990) e i metodi Markov Chain Monte Carlo (MCMC) (*Gibbs sampling*, *Metropolis sampling*, e *Hybrid Monte Carlo sampling*), basati sul campionamento stocastico (si veda Geman e Geman, 1984; Gilks *et al.*, 1996; MacKay, 1998).

Nel caso in cui non si disponga della struttura di una BN è richiesto l'*apprendimento strutturale*. Gli algoritmi per l'apprendimento strutturale delle Bayesian Network possono essere divisi in due famiglie.

ALGORITMI BASATI SU VINCOLI

Algoritmi che apprendono la struttura del grafo analizzando le relazioni probabilistiche derivanti dalla proprietà di Markov tramite test di indipendenza condizionale e costruendo un grafo che soddisfi le proprietà di *d-separazione*[3] corrispondenti. I modelli risultanti sono spesso interpretati come *modelli causali* (Pearl, 1988).

ALGORITMI BASATI SU PUNTEGGIO

Algoritmi che assegnano un punteggio (tramite una funzione di scoring) a tutte le strutture candidate e utilizzando tecniche di ottimizzazione cercano di raggiungere il punteggio massimo. Gli algoritmi di ricerca *greedy* (i. e. golosi, aggressivi o avidi a

3 Concetto di separazione direzionale tra insieme di nodi collegato al concetto di *indipendenza condizionale*. Ad esempio, quando un insieme di nodi $\mathbf{E}$ *d-separa* un insieme di nodi $\mathbf{X} = \{A, B\}$ allora A e B sono condizionalmente indipendenti dato $\mathbf{E}$.

seconda della traduzione preferita dall'inglese) sono la scelta più comune, tuttavia qualsiasi procedura di ricerca può essere usata.

Gli *algoritmi basati su vincoli* sono basati sull'algoritmo Inductive Causation (IC) di Verma e Pearl (1991) che fornisce un contesto teorico finalizzato all'apprendimento delle strutture dei modelli causali. L'algoritmo IC può essere riassunto nei tre passi successivi.

- Apprendimento dello scheletro (i. e. grafo non diretto) della rete. Poiché la ricerca esaustiva non è, nella maggior parte dei casi, computazionalmente realizzabile, tutti gli algoritmi di apprendimento restringono la ricerca al *Markov blanket*[4] di ogni nodo.
- Impostare la direzione degli archi che fanno parte di una *v-structure*[5].
- Impostare la direzione degli archi fra i nodi rimanenti affinché il vincolo di aciclicità sia rispettato.

Gli *algoritmi basati su punteggio* sono invece delle applicazioni dei vari algoritmi di ricerca euristica (e. g. *hill climbing*, *tabu search*, *best first search*, *simulated annealing*) che utilizzano una funzione di *scoring*. Solitamente la funzione di *scoring* utilizza la probabilità a posteriori dell'insieme dei dati di apprendimento (i. e. training set), data la struttura in esame, ed è *score-equivalent*, affinché reti che definiscono la stessa distribuzione di probabilità abbiano lo stesso score (Chickering, 2013). Tuttavia, per quanto questi algoritmi siano utilizzati molto frequentemente, essi sono esponenziali rispetto al numero di nodi della struttura del grafo. Inoltre, qualora si utilizzi una strategia di ricerca locale, è possibile che l'algoritmo restituisca come risultato un minimo locale (per evitare questa situazione si ricorre spesso a metodi di ricerca globale quali il MCMC). Si fa notare che è possibile ridurre il tempo necessario richiesto per l'apprendimento strutturale fissando un numero massimo di genitori candidati e cercando esaustivamente in insiemi di tale cardinalità una struttura che massimizzi l'informazione mutua fra variabili (Heckerman *et al.*, 1995).

1.1.2 Processi di Markov

Sempre al fine di preparare la discussione delle Continuos time Bayesian Network si prosegue presentando alcuni concetti propedeutici relativi ai processi di Markov, una categoria di processi stocastici con assenza di memoria (Loève, 1978).

4 Il *Markov blanket* di un nodo A è un insieme composto dai nodi genitori di A, dai suoi nodi figli e da tutti i nodi che condividono un figlio con A.

5 Una *v-structure* è una tripla di nodi $X_j \to X_i \leftarrow X_k$ incidenti su una connessione convergente.

Definizione 3 (Proprietà di Markov)**.** Secondo la proprietà di Markov gli stati futuri di un processo stocastico sono indipendenti dagli stati passati, avendo evidenza sullo stato presente di tale processo.

Formalmente, un processo stocastico X gode di tale proprietà, se e solo se vale la seguente equazione (Loève, 1978):

$$\mathbf{P}(X(t+\Delta t)\,|\,X(t),\,X(s)) = \mathbf{P}(X(t+\Delta t)\,|\,X(t)), \tag{1.3}$$

per ogni s, e t tali che $s < t < \infty$.

I modelli che rispettano tale proprietà sono detti modelli che rispettano l'*assunzione di Markov*.

Di conseguenza la distribuzione di probabilità condizionale degli stati futuri di un processo stocastico che gode di tale proprietà è indipendente dagli stati passati dato quello attuale.

In altri termini ciò indica che lo stato futuro di una variabile casuale è *condizionalmente indipendente* (si veda la definizione 2) dalla sequenza dei suoi stati passati, avendo evidenza sul suo stato presente.

Dalla proprietà di Markov deriva la definizione dei processi di Markov.

Definizione 4 (Processo di Markov)**.** Si definisce (Loève, 1978) come processo di Markov un processo stocastico che gode della proprietà di Markov.

Definizione 5 (Catena di Markov)**.** Un processo di Markov che può assumere solo un numero finito di stati è solitamente definito come una *catena di Markov* (si veda Norris, 1998, p. 10).

Esistono due tipi di processi di Markov: omogenei e non. Si procede quindi fornendone le definizioni.

Definizione 6 (Processo di Markov omogeneo)**.** Un processo di Markov è detto *omogeneo* qualora $\mathbf{P}(X(t+\Delta t)\,|\,X(t))$ non dipenda dal tempo t. Affinché ciò sia vero, ponenedo $t = 0$, deve risultare che:

$$\mathbf{P}(X(t+\Delta t)\,|\,X(t)) = \mathbf{P}(X(\Delta t)\,|\,X(0)). \tag{1.4}$$

Data quindi una variabile casuale X e l'insieme delle sue istanziazioni $\mathrm{val}(X) = \{x_1, \ldots, x_J\}$, $X(t)$ è un processo di Markov *omogeneo, a tempo continuo e stati finiti* se e solo se la sua dinamica è definibile in termini di:

- una distribuzione di probabilità iniziale $\mathbf{P}^0_X$ su $\mathrm{val}(X)$
- una matrice di intensità $\mathbf{Q}_X$.

Definizione 7 (Matrice di intensità)**.** Una matrice di intensità (IM), rappresenta un *modello di transizione Markoviano*:

$$\mathbf{Q}_X = \begin{bmatrix} -\mathbf{q}_{x_1} & \mathbf{q}_{x_1 x_2} & \cdots & \mathbf{q}_{x_1 x_K} \\ \mathbf{q}_{x_2 x_1} & -\mathbf{q}_{x_2} & \cdots & \mathbf{q}_{x_2 x_K} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{q}_{x_K x_1} & \mathbf{q}_{x_K x_2} & \cdots & -\mathbf{q}_{x_K} \end{bmatrix}.$$

Lo scopo di un matrice di intensità è descrivere il comportamento transiente di X, un processo di Markov omogeneo.

Osservazione 7.1. L'*ordine* di una matrice di intensità corrisponde a $K = |val(X)|$, la cardinalità dell'insieme dei valori assunti da X.

Affinché $\mathbf{Q}_X$ sia una matrice di intensità valida, ogni sua riga deve sommare a 0:

$$\mathbf{q}_{x_i} = \sum_{i \neq j} \mathbf{q}_{x_i x_j} \quad \text{con} \quad \mathbf{q}_{x_i}, \mathbf{q}_{x_i x_j} > 0.$$

Data quindi una matrice di intensità $\mathbf{Q}_X$ essa descrive il comportamento transiente di $X(t)$. Se $X(0) = x_i$, allora il processo di Markov omogeneo (e indicizzato dal tempo t) $X(t)$ rimarrà nello stato x_i una quantità di tempo *esponenzialmente distribuita* rispetto al parametro $\mathbf{q}_{x_i}$. Di conseguenza la *funzione di densità* f e la corrispondente *funzione di ripartizione*[6] F sono:

$$\begin{aligned} f(t) &= \mathbf{q}_{x_i} e^{-\mathbf{q}_{x_i} t}, \quad t > 0 \\ F(t) &= 1 - e^{-\mathbf{q}_{x_i} t}, \quad t \geqslant 0. \end{aligned} \tag{1.5}$$

Quando un modello di transizione è definito esclusivamente tramite una matrice di intensità $\mathbf{Q}_X$ si dice che esso usa una *parametrizzazione pura* delle intensità. In tal caso i parametri per un processo di Markov omogeneo con K stati sono $\{\mathbf{q}_{x_i}, \mathbf{q}_{x_i x_j} : 1 \leqslant i, j \leqslant K, i \neq j\}$.

Mentre gli elementi sulla diagonale di una matrice di intensità, $\mathbf{q}_{x_i}$, codificano una quantità che può essere interpretata come la *"probabilità istantanea"* che X abbandoni lo stato x_i, gli elementi non sulla diagonale, $\mathbf{q}_{x_i x_j}$, esprimono l'*intensità di transizione* dallo stato x_i allo stato x_j.

Tuttavia, questa non è l'unica parametrizzazione possibile per un processo di Markov omogeneo. Si noti infatti che la distribuzione di probabilità locale sulle transizioni di X è fattorizzata in due parti.

6 Nel calcolo delle probabilità la funzione di ripartizione di una variabile casuale X a valori reali, anche nota come funzione di distribuzione cumulativa, è la funzione che associa a ciascun valore x la probabilità che X assuma valori minori o uguali ad x.

Definizione 8 (Parametrizzazione mista delle intensità)**.** La *parametrizzazione mista* delle intensità per un processo di Markov omogeneo X con K stati è composta da due insiemi di parametri:

$$\mathbf{q}_X = \{\, \mathbf{q}_{x_i} : 1 \leqslant i \leqslant K \,\},$$
$$\theta_X = \{\, \theta_{x_i x_j} : 1 \leqslant i, j \leqslant K, i \neq j \,\}.$$

La semantica di tali insiemi di parametri è la seguente:

- $\mathbf{q}_X$ è un insieme di intensità $\mathbf{q}_{x_i}$ che parametrizzano una *distribuzione di probabilità esponenziale* ed esprimono quando avvengono le transizioni
- θ_X è un insieme di probabilità $\theta_{x_i x_j}$ che esprimono la *distribuzione di probabilità multinomiale* tra coppie di stati differenti $i \neq j$.

Osservazione 8.1. Si osservi che, aldilà del tipo di parametrizzazione con cui si sceglie di definire un modello di transizione, il numero di parametri necessari è pari a K^2 sebbene il numero di parametri liberi sia solo $K^2 - K$ (Nodelman, 2007).

Osservazione 8.2. Si noti, inoltre, che una parametrizzazione può essere più conveniente dell'altra a seconda del processo in cui si è coinvolti, di conseguenza nel prosieguo le si utilizzerà entrambe in modo intercambiabile.

Al fine di correlare questi due tipi di parametrizzazione dei modelli di transizione si riporta il seguente teorema (Nodelman, 2007).

Teorema 1.2. *Dati X e Y, due processi di Markov omogenei con lo stesso spazio degli stati e la stessa distribuzione di probabilità iniziale, se il modello di transizione di X è definito tramite la matrice di intensità $\mathbf{Q}_X$ e quello di Y è definito tramite la parametrizzazione mista $\boldsymbol{q}_Y$, θ_Y, allora X e Y sono stocasticamente equivalenti*[7] *solo se:*

$$\boldsymbol{q}_{y_i} = \boldsymbol{q}_{x_i}$$

e

$$\theta_{y_i y_j} = \frac{\boldsymbol{q}_{x_i x_j}}{\boldsymbol{q}_{x_i}}.$$

Osservazione 8.3. Si osservi che il teorema 1.2 formalizza la relazione che sussiste fra i parametri $\mathbf{q}$ e θ.

Quindi, qualsiasi sia la parametrizzazione utilizzata per rappresentare il modello di transizione di un processo di Markov omogeneo X, è possibile calcolare:

7 Due processi di Markov sono detti *stocasticamente equivalenti* se posseggono lo stesso spazio degli stati e le stesse probabilità di transizione (Gihman e Skorohod, 1973).

- il *tempo atteso* di una transizione uscente dallo stato x_i

$$1/\mathbf{q}_{x_i}$$

- la *"probabilità istantanea"* di transizione dallo stato x_i allo stato x_j

$$\theta_{x_i x_j} = \mathbf{q}_{x_i x_j}/\mathbf{q}_{x_i}.$$

Infine, si noti che la matrice $\mathbf{Q}_X$ fa in modo che X soddisfi la proprietà di Markov poiché il comportamento futuro di X è definito solamente in base al suo stato attuale (vale l'equazione 1.4).

Definizione 9 (Processo di Markov condizionale)**.** Un processo di Markov le cui intensità di transizione variano nel tempo non in funzione del tempo ma in funzione dei valori assunti ad ogni determinato istante t da un insieme di altre variabili, che evolvono anch'esse come dei processi di Markov, è detto essere un processo di Markov condizionale (o processo di Markov non omogeneo).

Assumendo quindi che una variabile casuale X evolva come un processo di Markov $X(t)$ e che la sua dinamica sia condizionata da un insieme di altre variabili casuali $Pa(X)$, anch'esse dei processi di Markov, è possibile definire per tale variabile casuale una matrice di intensità condizionale (CIM) $\mathbf{Q}_{X|Pa(X)}$.

Specificando una distribuzione di probabilità iniziale su X si definisce quindi un processo di Markov il cui comportamento dipende dalle istanziazioni dei valori di $Pa(X)$.

Definizione 10 (Matrice di intensità condizionale)**.** Dato un insieme di processi di Markov $Pa(X)$, una matrice di intensità condizionale $\mathbf{Q}_{X|Pa(X)}$ è costituita da un insieme di matrici di intensità $\mathbf{Q}_{X|pa_i(x)}$, una per ogni diversa istanziazione $pa_i(x)$ di $Pa(X)$ (Stella e Amer, 2012):

$$\mathbf{Q}_{X|Pa(X)} = \left\{ \mathbf{Q}_{X|pa_1(x)}, \mathbf{Q}_{X|pa_2(x)}, \dots, \mathbf{Q}_{X|pa_n(x)} \right\}.$$

Ogni matrice di intensità di $\mathbf{Q}_{X|Pa(X)}$ è del seguente tipo:

$$\mathbf{Q}_{X|pa_i(x)} = \begin{bmatrix} -\mathbf{q}^{pa_i(x)}_{x_1} & \mathbf{q}^{pa_i(x)}_{x_1 x_2} & \cdots & \mathbf{q}^{pa_i(x)}_{x_1 x_K} \\ \mathbf{q}^{pa_i(x)}_{x_2 x_1} & -\mathbf{q}^{pa_i(x)}_{x_2} & \cdots & \mathbf{q}^{pa_i(x)}_{x_2 x_K} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{q}^{pa_i(x)}_{x_K x_1} & \mathbf{q}^{pa_i(x)}_{x_K x_2} & \cdots & -\mathbf{q}^{pa_i(x)}_{x_K} \end{bmatrix}.$$

Di seguito si presenta un breve esempio finalizzato alla comprensione pratica delle matrici di intensità condizionali (CIM) e del loro scopo.

Esempio 10.1.
Date due variabili causali, $E(t)$ e $H(t)$, delle quali la prima modella l'eventualità che un individuo stia mangiando o meno (se $e = 2$ allora l'individuo sta mangiando, viceversa se $e = 1$) mentre la seconda modella l'eventualità che lo stesso individuo abbia fame o meno (se $h = 2$ allora l'individuo è affamato, viceversa se $h = 1$) e la matrice di intensità condizionale $\mathbf{Q}_{E \mid H}$, che è un insieme composto dalle matrici di intensità $\mathbf{Q}_{E \mid h=1}$ e $\mathbf{Q}_{E \mid h=2}$, è possibile calcolare la probabilità degli eventi della variabile casuale E condizionatamente all'evidenza che si possiede sulla variabile casuale H.

$$\mathbf{Q}_{E \mid h=1} = \begin{array}{c} \\ 1 \\ 2 \end{array} \begin{array}{c} \begin{array}{cc} 1 & 2 \end{array} \\ \begin{bmatrix} -.01 & .01 \\ 10 & -10 \end{bmatrix} \end{array} \qquad \mathbf{Q}_{E \mid h=2} = \begin{array}{c} \\ 1 \\ 2 \end{array} \begin{array}{c} \begin{array}{cc} 1 & 2 \end{array} \\ \begin{bmatrix} -2 & 2 \\ .01 & -.01 \end{bmatrix} \end{array}.$$

Ipotizzando che l'unità temporale corrisponda a un'ora:

- un individuo affamato ($h = 2$) che non sta mangiando ($e = 1$) inizierà a mangiare in 30 minuti poiché

$$\frac{1}{\mathbf{q}_{e=1 \mid h=2}} = \frac{1}{2},$$

- un individuo non affamato ($h = 1$) che sta mangiando ($e = 2$) smetterà di mangiare ($e = 1$) entro 6 minuti poiché

$$\frac{1}{\mathbf{q}_{e=2 \mid h=1}} = \frac{1}{10};$$

si osservi che la *"probabilità istantanea"* di transizione da $e = 2$ a $e = 1$ è

$$\theta_{e=2,e=1 \mid h=1} = \frac{\mathbf{q}_{e=2,e=1 \mid h=1}}{\mathbf{q}_{e=2 \mid h=1}} = \frac{10}{10} = 1,$$

ciò poiché $val(E) = 2$, il ché implica che le matrici di intensità sono matrici 2×2 e, dovendo ogni loro riga sommare a 0, gli elementi sulla diagonale sono uguali al rispettivo (stessa riga) e unico elemento non sulla diagonale.

1.2 DEFINIZIONI PRELIMINARI

Nelle precedenti sezioni sono stati illustrati i concetti che si pongono a fondamento delle Continuos time Bayesian Network:

- le Bayesian Network: utili a comprendere la rappresentazione strutturata dello spazio degli stati delle CTBN, l'utilizzo della nozione di indipendenza condizionale e le conseguenti tecniche di apprendimento e inferenza

- i processi di Markov, omogenei e non, al fine di introdurre le modalità di rappresentazione (qualitativa e quantitativa) delle CTBN.

Prima di presentare le Continuos time Bayesian Network come una collezione di processi di Markov a tempo continuo non omogenei e con spazio degli stati discreto (Nodelman, 2007), si forniscono alcune definizioni utili per il prosieguo della discussione.

Definizione 11 (Variabile di processo)**.** Una variabile di processo $\mathbf{X}$, anche detta Process Variable (PV) (Nodelman, 2007), è un insieme di processi di Markov a tempo continuo $X(t)$.

Definizione 12 (Traiettoria)**.** Istanziazione di un insieme di valori per $X(t)$ al variare di t.

Definizione 13 (J-time-segment)**.** Partizionamento di un intervallo temporale $[0, T)$ in J intervalli chiusi a sinistra:

$$[0, t_1)\,;\,[t_1, t_2)\,;\,\ldots\,;\,[t_{J-1}, T).$$

Nota 13.1. È possibile riferirsi a tale concetto anche tramite l'espressione *"insieme dei segmenti temporali"*.

Definizione 14 (J-evidence-stream)**.** Data una variabile di processo $\mathbf{X}$ composta da N variabili casuali e un insieme di segmenti temporali composto da J intervalli, un J-*evidence-stream* è l'insieme delle istanziazioni comuni $\mathbf{X} = \mathbf{x}$ associate ad ogni intervallo temporale per ogni sottoinsieme delle variabili casuali (Stella e Amer, 2012). È denotato con $(\mathbf{X}^1 = \mathbf{x}^1\,, \mathbf{X}^2 = \mathbf{x}^2\,, \ldots\,, \mathbf{X}^J = \mathbf{x}^J)$, o più concisamente con $(\mathbf{x}^1\,, \mathbf{x}^2\,, \ldots\,, \mathbf{x}^J)$.

Nota 14.1. È possibile riferirsi a tale concetto anche tramite l'espressione *"flusso di evidenze"*.

Nota 14.2. Un flusso di evidenze $(\mathbf{x}^1\,, \mathbf{x}^2\,, \ldots\,, \mathbf{x}^J)$. è detto essere *completamente osservato* se lo stato di tutte le variabili $X_n \in \mathbf{X}$ è conosciuto in tutto l'intervallo $[0, T)$. Viceversa, un flusso di evidenze è detto *parzialmente osservato*.

1.3 RAPPRESENTAZIONE

Una Continuos time Bayesian Network è un modello grafico in cui ogni nodo rappresenta una variabile casuale i cui stati evolvono in modo continuo bel tempo. Le dinamiche evolutive degli stati dei nodi sono governate e dipendono dal valore che gli stati dei nodi padre[8] assumono (Stella e Amer, 2012). Quindi ogni nodo è un processo di

8 Con il termine "nodo padre", o *parent node*, si intende un nodo il cui stato condiziona quello di un altro nodo del modello grafico.

Markov condizionale (si veda la definizione 9) a tempo continuo e spazio degli stati discreto.

Una CTBN è composta principalmente da due componenti:

- una distribuzione di probabilità iniziale
- le dinamiche che regolano l'evoluzione nel tempo continuo della distribuzione di probabilità

Più formalmente si definisce:

Definizione 15 (Continuos time Bayesian Network)**.** Data una variabile di processo $\mathbf{X}$, insieme di processi di Markov $X_1, X_2, \ldots, X_N$ a tempo continuo e con spazio degli stati finito $val(X_n) = \{x_1, \ldots, x_J\}$ (dove $n = 1, \ldots, N$), una CTBN $\mathcal{N}$ su $\mathbf{X}$ consiste di:

- una distribuzione di probabilità iniziale $\mathbf{P}^0_{\mathbf{X}}$ specificata come una Bayesian Network $\mathcal{B}$ su $\mathbf{X}$
- un modello di transizione a tempo continuo, specificato da:
  - un grafo $\mathcal{G}$, orientato e non necessariamente aciclico, composto dai nodi $X_1, X_2, \ldots, X_N$, ognuno dei quali possiede un insieme di genitori denotato da $Pa(X_n)$
  - una matrice di intensità condizionale $\mathbf{Q}_{X_n \mid Pa(X_n)}$ per ogni nodo $X_n \in \mathbf{X}$.

Per ogni variabile causale $X_n \in \mathbf{X}$ di $\mathcal{N}$ si ha quindi un insieme di modelli di probabilità locali: $\mathbf{Q}_{X_n \mid Pa(X_n)}$, la CIM di X_n, è infatti un insieme di modelli di transizione Markoviani la cui cardinalità è pari a quella dell'insieme delle diverse istanziazioni di $Pa(X_n)$.

Si riscontra, quindi, quanto già affermato in precedenza (si veda 1.1), cioè che una CTBN esprime la sua dinamica evolutiva globale tramite un unico processo di Markov omogeneo, costituito da un insieme di processi di Markov condizionali (un insieme di CIM e relative distribuzioni di probabilità iniziali).

Si noti che, diversamente dalle Bayesian Network, nelle Continuos time Bayesian Network gli archi fra i nodi rappresentano le dipendenze nel tempo. Per tale motivo è possibile che la componente $\mathcal{G}$ del modello di transizione continuo contenga dei cicli. Tra l'altro, come vedremo nel prosieguo, la mancanza di tale vincolo di aciclicità porta a notevoli vantaggi computazionali relativamente all'apprendimento della struttura di una CTBN dai dati.

1.4 APPRENDIMENTO

In questa sezione si argomenta sulla probabilità di un *insieme di dati completo* rispetto a una Continuos time Bayesian Network. A tal fine si mostra come una CTBN, essendo un modello esponenziale, possa

essere decomposta in un aggregato di modelli di probabilità locali relativi alle singole variabili casuali e espressa in termini di *statistiche sufficienti* aggregate.

Si affronta infine il processo di apprendimento dei parametri delle Continuos time Bayesian Network da *dati completi*. I processi di apprendimento relativi a dati non completi sono tralasciati poiché non facenti parte degli argomenti di questo lavoro di tesi.

Definizione 16 (Insieme di dati completo)**.** Dato un insieme di variabili casuali, un insieme di dati $\mathcal{D} = \{ \delta_1 , \dots , \delta_h \}$ si dice *completo* se ogni δ_i (con $i = 1 , \dots , h$) è un insieme di traiettorie completamente osservate delle variabili casuali (i. e. l'istanziazione di tutte le variabili casuali è osservabile per ogni istante temporale di ogni traiettoria).

1.4.1 Statistiche sufficienti

Le *statistiche sufficienti* per un singolo processo di Markov omogeneo $X(t)$ riassumono la sua dinamica evolutiva con:

- $T[x]$: la quantità di tempo trascorsa nello stato x
- $M[x , x']$: il numero di transizioni dallo stato x allo stato x'.

Il numero totale di transizioni uscenti da uno stato x è:

$$M[x] = \sum_{x'} M[x , x'].$$

Nel caso di un processo di Markov condizionale è invece necessario considerare anche l'istanziazione dell'insieme $Pa(X)$ dei nodi genitori:

- $T[x | pa_i(x)]$: la quantità di tempo trascorsa nello stato x quando $Pa(X) = pa_i(x)$
- $M[x , x' | pa_i(x)]$: il numero di transizioni dallo stato x allo stato x' quando $Pa(X) = pa_i(x)$.

Chiaramente, il numero totale di transizioni si calcola come sopra.

1.4.2 Likelihood

Al fine di presentare il calcolo della likelihood[9] di una CTBN rispetto a un dataset completo $\mathcal{D}$ è bene procedere per gradi e iniziare presentando dapprima la likelihood di una singola transizione di un singolo processo di Markov omogeneo $X(t)$.

9 La likelihood di un insieme di valori per i parametri, dato un insieme di dati, è uguale alla probabilità dei dati, dati tali valori per i parametri.

Likelihood di una singola transizione

Data una tripla $d = \langle x_d , t_d , x_{d'} \rangle \in \mathcal{D}$, la quale esprime una transizione di $X(t)$ da x_d a $x_{d'}$ dopo che esso ha trascorso t_d tempo in x_d, è possibile scrivere la likelihood di questa singola transizione d in funzione dei parametri (1.4.3):

$$\begin{aligned} L_X(\mathbf{q} , \theta : d) &= L_X(\mathbf{q} : d) \cdot L_X(\theta : d) \\ &= \mathbf{q}_{x_d} \, e^{-\mathbf{q}_{x_d} t_d} \cdot \theta_{x_d x_{d'}}. \end{aligned} \tag{1.6}$$

Si noti che l'equazione 1.6 è ricavata moltiplicando la *funzione di distribuzione di probabilità* di $X(t)$ (equazione 1.5) per la *"probabilità istantanea"* di transizione (si veda la definizione 7).

Likelihood di un dataset completo

Poiché tutte le transizioni sono osservabili, la *likelihood* del dataset $\mathcal{D}$ può essere decomposta come un prodotto delle likelihood individuali di ogni singola transizione d (si veda Nodelman *et al.*, 2002, p. 3). Per tale motivo $\mathcal{D}$ è sintentizzabile aggregando le *statistiche sufficienti* relative a ogni processo di Markov condizionale di una CTBN.

Quindi la likelihood di un dataset completo $\mathcal{D}$ rispetto a un singolo processo di Markov omogeneo $X(t)$ è:

$$\begin{aligned} L_X(\mathbf{q} , \theta : \mathcal{D}) &= \Big[\prod_{d \in \mathcal{D}} L_X(\mathbf{q} : d) \Big] \Big[\prod_{d \in \mathcal{D}} L_X(\theta : d) \Big] \\ &= \Big[\prod_x (\mathbf{q}_x)^{M[x]} e^{-\mathbf{q}_x T[x]} \Big] \Big[\prod_x \prod_{x \neq x'} (\theta_{xx'})^{M[x,x']} \Big]. \end{aligned} \tag{1.7}$$

Si supponga ora di traslare questo concetto a una Continuos time Bayesian Network $\mathcal{N}$ con N nodi: per ogni nodo X_i, con $i = 1 , \dots N$ è necessario considerare tutte le transizioni contestualmente all'istanziazione dell'insieme $Pa(X_i)$ dei suoi nodi genitori. Poiché, nel caso di *dati completi*, si conosce sempre l'istanziazione di $Pa(X_i)$, allora, per ogni istante di tempo t, si conosce quale matrice di intensità $\mathbf{Q}_{X_i \mid pa_i(x)}$, con $pa_i(x) \in Pa(X_i)$, governi la dinamica di X_i.

Perciò la probabilità dei dati $\mathcal{D}$ rispetto a $\mathcal{N}$ è il prodotto delle likelihood di ogni variabile X_i:

$$\begin{aligned} L_{\mathcal{N}}(\mathbf{q} , \theta : \mathcal{D}) &= \prod_{X_i \in \mathbf{X}} L_{X_i}(\mathbf{q}_{X_i \mid Pa(X_i)} , \theta_{X_i \mid Pa(X_i)} : \mathcal{D}) \\ &= \prod_{X_i \in \mathbf{X}} L_{X_i}(\mathbf{q}_{X_i \mid Pa(X_i)} : \mathcal{D}) \, L_{X_i}(\theta_{X_i \mid Pa(X_i)} : \mathcal{D}). \end{aligned} \tag{1.8}$$

Il termine $L_X(\theta_{X \mid Pa(X)} : \mathcal{D})$ esprime la likelihood delle transizioni tra stati. Si osservi, inoltre, come il tempo che intercorre fra le transizioni sia trascurato poiché esse dipendono esclusivamente dal valore di

nodi genitori (si veda Nodelman *et al.*, 2002, p. 3). Quindi, usando le *statistiche sufficienti* si può scrivere:

$$L_X(\theta_{X|Pa(X)} : \mathcal{D}) = \prod_{pa_i(x)} \prod_{x} \prod_{x \neq x'} (\theta_{xx'|pa_i(x)})^{M[x,x'|pa_i(x)]}. \quad (1.9)$$

Per quanto riguarda il calcolo di $L_X(q_{X|Pa(X)} : \mathcal{D})$ va considerato il caso in cui il tempo trascorso da X in uno determinato stato x termini non a causa di una sua transizione bensì a causa di una transizione di uno o più nodi appartenenti all'insieme dei suoi nodi genitori (i. e. una nuova istanziazione per l'insieme dei genitori $Pa(X)$). È quindi necessario considerare la probabilità che il nodo X rimanga in x una quantità di tempo *almeno pari* a t mentre i suoi nodi genitori $Pa(X)$ non effettuano alcuna transizione di stato (si veda Nodelman *et al.*, 2002, p. 3). Tale quantità si ricava dalla funzione di distribuzione cumulativa di una distribuzione esponenziale (equazione 1.5):

$$1 - F(t) = e^{-q_{x|pa_i(x)}t}.$$

Perciò la likelihood delle quantità di tempo trascorse in ogni stato è:

$$L_X(q_{X|Pa(X)} : \mathcal{D}) = \prod_{pa_i(x)} \prod_{x} (q_{x|pa_i(x)})^{M[x|pa_i(x)]} e^{-q_{x|pa_i(x)}T[x|pa_i(x)]}. \quad (1.10)$$

Combinando l'equazione 1.10 e l'equazione 1.9 si ottiene la likelihood di un dataset completo $\mathcal{D}$ rispetto a un singolo processo di Markov condizionale:

$$L_X(q, \theta : \mathcal{D}) = \prod_{pa_i(x)} \prod_{x} \Big[(q_{x|pa_i(x)})^{M[x|pa_i(x)]} e^{-q_{x|pa_i(x)}T[x|pa_i(x)]} \cdot \prod_{x \neq x'} (\theta_{xx'|pa_i(x)})^{M[x,x'|pa_i(x)]} \Big]. \quad (1.11)$$

Si noti che, dal punto di vista computazionale, è conveniente riformulare l'equazione 1.11 come *log-likelihood*:

$$\ell_X(q, \theta : \mathcal{D}) = \sum_{pa_i(x)} \sum_{x} \Big[M[x|pa_i(x)] \ln(q_{x|pa_i(x)}) - q_{x|pa_i(x)} T[x|pa_i(x)] + \sum_{x \neq x'} M[x, x'|pa_i(x)] \ln(\theta_{xx'|pa_i(x)}) \Big]. \quad (1.12)$$

È ora possibile asserire che la *log-likelihood* di $\mathcal{N}$ (dall'equazione 1.8) è:

$$\ell_{\mathcal{N}}(q, \theta : \mathcal{D}) = \sum_{X_i \in \mathbf{X}} \ell_{X_i}(q, \theta : \mathcal{D}). \quad (1.13)$$

In questa sezione si è presentato come computare la likelihood di un modello di una CTBN rispetto a un dataset completo.

Tuttavia, nel caso in cui non si conoscano i parametri di una CTBN è necessario stimarli. Nella prossima sezione viene affrontato esattamente questo argomento.

1.4.3 Stima dei parametri

Si affronta ora il problema dell'apprendimento dei parametri di una Continuos time Bayesian Network (con struttura nota $\mathcal{G}$) da un insieme di dati completi di tipo *multinomiale* (si veda Nodelman, 2007, sezione 5.1).

Quando si tratta con dati multinomiali ci sono principalmente due scelte che è possibile fare. La scelta più semplice consiste nell'effettuare una stima dei parametri del modello tramite un approccio *maximum-likelihood*. Tuttavia, è noto che tale approccio può portare a problemi con l'inferenza quando i dati di input sono sparsi. Per evitare tale limitazione solitamente si effettua una *regolarizzazione bayesiana* dei parametri: si sceglie una distribuzione a priori per i parametri e li si aggiorna in accordo ai dati di input.

Come costruire una CIM dai parametri.

La stima dei parametri non è un processo fine a se stesso, in quanto, da essi è possibile costruire le matrici di intensità condizionali (CIM) di ogni nodo della CTBN. Come si ricorderà, una CIM è un insieme di matrici di intensità, una per ogni istanziazione $pa_i(x)$ dei nodi genitori (si veda la definizione 10). Perciò, fissato $pa_i(x)$, si può computare la rispettiva matrice di intensità per un nodo qualsiasi ponendo sulla diagonale il rispettivo vettore dei parametri $\mathbf{q}_{x\,|\,pa_i(x)}$ e ricavando i valori non sulla diagonale dalla relazione (si veda il teorema 1.2) fra i parametri $\mathbf{q}$ e θ:

$$\mathbf{q}_{xx'\,|\,pa_i(x)} = \theta_{xx'\,|\,pa_i(x)} \cdot \mathbf{q}_{x\,|\,pa_i(x)}. \tag{1.14}$$

Infine, come vedremo in seguito nel capitolo 3, i parametri sono anche un componente chiave del processo di apprendimento strutturale.

Stima maximum-likelihood

In base a quanto attestato dalla definizione stessa delle CTBN (15), la dinamica evolutiva globale di una CTBN, cioè la dinamica di tutti i nodi di $\mathcal{G}$ (dei processi di Markov condizionali indicizzati dal tempo), è espressa tramite un processo di Markov omogeneo. Dalla definizione 7, inoltre, si deduce che tale processo di Markov induce un modello di probabilità composto da una *distribuzione esponenziale* con parametro $\mathbf{q}_{x\,|\,pa_i(x)}$, che esprime il tempo trascorso in uno stato x da un nodo X data una istanziazione $pa_i(x)$ per i nodi genitori $Pa(X)$, e una *distribuzione multinomiale* con parametro $\theta_{xx'\,|\,pa_i(x)}$, che esprime il numero di transizioni uscenti da uno stato x verso x' (sempre fermo restando il condizionamento dato dall'istanziazione dei nodi genitori).

La media della distribuzione esponenziale in questione è pari a $1/\mathbf{q}_{x\,|\,pa_i(x)}$. Questa quantità esprime il tempo medio delle transizioni uscenti da uno stato x, fermo restando che il genitore del nodo in questione abbia istanziazione costante e uguale a $pa_i(x)$. Poiché il tempo medio si calcola rapportando il tempo totale trascorso in

x, $T[x \mid pa_i(x)]$, rispetto al numero totale di transizioni uscenti da x, $M[x \mid pa_i(x)]$, si ottiene:

$$\frac{1}{\mathbf{q}_{x \mid pa_i(x)}} = \frac{T[x \mid pa_i(x)]}{M[x \mid pa_i(x)]}.$$

Invece, la probabilità di transizione da uno stato x verso x' è data dal rapporto tra il numero totale di transizioni da x a x' diviso il numero totale di transizioni uscenti da x; cioè:

$$\frac{M[x, x' \mid pa_i(x)]}{M[x \mid pa_i(x)]}.$$

Teorema 1.3. *Parametri maximum-likelihood (MLE). I parametri che massimizzano la likelihood (equazione 1.13) di una Continuos time Bayesian Network sono funzione delle statistiche sufficienti:*

$$\begin{aligned} \mathbf{q}_{x \mid pa_i(x)} &= \frac{M[x \mid pa_i(x)]}{T[x \mid pa_i(x)]} \\ \theta_{xx' \mid pa_i(x)} &= \frac{M[x, x' \mid pa_i(x)]}{M[x \mid pa_i(x)]}. \end{aligned} \tag{1.15}$$

Si noti che, in questo caso (*dataset completo*), $\mathbf{q}_{x \mid pa_i(x)}$ e $\theta_{xx' \mid pa_i(x)}$ sono delle *stime esatte*. Essi massimizzano la probabilità a posteriori di un dataset, dato un modello CTBN.

Stima bayesiana

Un approccio alternativo alla stima dei parametri è la stima bayesiana (si veda Nodelman, 2007, sezione 5.1.1).

A tal fine è necessario definire una distribuzione a priori sui parametri di una CTBN. Come si è soliti fare in situazioni di questo tipo, per tale distribuzione si sceglie di usare una *distribuzione a priori coniugata*[10] poiché ciò risulta conveniente dal punto di vista algebrico (e quindi computazionale). Infatti, una distribuzione a priori coniugata fornisce un'espressione in forma chiusa per la distribuzione a posteriori (alternativamente potrebbe risultare necessario il calcolo di un integrale numerico).

Si consideri innanzitutto un singolo processo di Markov. Si ricorda (si vedano a tal riguardo le definizioni 7 e 8) che un processo di Markov ha due insiemi di parametri: θ che parametrizzano una *distribuzione multinomiale* e $\mathbf{q}$ che parametrizzano una *distribuzione esponenziale*.

10 In teoria della probabilità bayesiana, se le distribuzioni a posteriori $P(\theta|x)$ sono nella stessa famiglia della distribuzione a priori $P(\theta)$, le due distribuzioni sono definite coniugate, e la distribuzione a priori è chiamata *distribuzione a priori coniugata* per la verosimiglianza (*likelihood*). Una distribuzione a priori coniugata è conveniente dal punto di vista algebrico in quanto fornisce una espressione in forma chiusa per la distribuzione a posteriori e perché può fornire delle intuizioni circa il modo con cui la funzione di verosimiglianza aggiorna la distribuzione.

Una distribuzione a priori coniugata per il parametro $\mathbf{q}$ è la *distribuzione Gamma* $P(\mathbf{q}) = \mathrm{Gamma}(\alpha_x, \tau_x)$, dove (si veda Nodelman, 2007):

$$P(\mathbf{q}) = \frac{\tau_x^{\alpha_{x+1}}}{\Gamma(\alpha_{x+1})} \mathbf{q}^{\alpha_x} e^{-\mathbf{q}\tau_x}. \tag{1.16}$$

Invece, avendo assunto l'indipendenza dei parametri e poiché la funzione di densità della distribuzione di probabilità di θ, che è una multinomiale, è positiva, per essa si sceglie come priori coniugata la *distribuzione di Dirichlet* $P(\theta) = \mathrm{Dir}(\alpha_{xx_1}, \ldots, \alpha_{xx_K})$ (si veda Heckerman, 1996; Heckerman *et al.*, 1995), la cui funzione di densità (Steck, Harald and Jaakkola, 2002) è:

$$P(\theta) = \frac{\Gamma(\alpha_x)}{\Gamma(\alpha_{xx_1}) \cdot \ldots \cdot \Gamma(\alpha_{xx_K})} \theta_{xx_k}^{\alpha_{xx_1}-1} \cdot \ldots \cdot \theta_{xx_1}^{\alpha_{xx_K}-1}. \tag{1.17}$$

Nota 1.4.1. Si noti che l'iper-parametro α_x, detto *dimensione equivalente del campione*, è costituito dalla somma dei *conteggi immaginari* $\alpha_{xx_1} + \ldots + \alpha_{xx_K}$, chiamati anche *pseudo-conteggi* (Steck, Harald and Jaakkola, 2002). Esso può essere pensato come un fattore che esprime la *"forza"* della distribuzione a priori, in quanto, più esso aumenta, più le stime dei parametri sono regolarizzate, cioè meno estreme. Chiaramente, quando α_x tende a 0 le stime dei parametri tendono alle stime maximum-likelihood. In letteratura (si veda Steck, Harald and Jaakkola, 2002) questo processo è anche chiamato *"smoothing"*.

Nota 1.4.2. L'iper-parametro τ_x, invece, rappresenta una *quantità di tempo immaginaria* che incorpora la credenza della distribuzione a priori sul parametro della distribuzione esponenziale.

Quindi, se si assume che i parametri sono *stocasticamente indipendenti*, cioè che $P(\theta, \mathbf{q}) = P(\theta)\,P(\mathbf{q})$, allora le distribuzioni a posteriori (i. e. condizionate sui dati) dei parametri $\mathbf{q}$ e θ sono:

$$\begin{aligned} P(\mathbf{q} \mid \mathcal{D}) &= \mathrm{Gamma}(\alpha_x + M[x], \tau_x + T[x]) \\ P(\theta \mid \mathcal{D}) &= \mathrm{Dir}(\alpha_{xx_1} + M[x, x_1], \ldots, \alpha_{xx_K} + M[x, x_K]). \end{aligned} \tag{1.18}$$

Al fine di generalizzare quest'idea e ottenere una distribuzione a priori coniugata per un'intera CTBN è necessario che essa soddisfi due assunzioni (comuni per le distribuzioni a priori nelle Bayesian Network, si veda Heckerman (1996)): l'indipendenza globale e locale dei parametri. In base all'*indipendenza globale dei parametri* si può scrivere:

$$P(\mathbf{q}, \theta) = \prod_{X_i \in \mathbf{X}} P(\mathbf{q}_{X_i \mid Pa(X_i)}, \theta_{X_i \mid Pa(X_i)}). \tag{1.19}$$

Invece, dall'*indipendenza locale dei parametri* consegue che è possibile scrivere:

$$P(\mathbf{q}_{X \mid Pa(X)}, \theta_{X \mid Pa(X)}) = \left[\prod_x \prod_{pa_i(x)} P(\mathbf{q}_{x \mid pa_i(x)})\right]\left[\prod_x \prod_{pa_i(x)} P(\theta_{x \mid pa_i(x)})\right]. \tag{1.20}$$

Se tale distribuzione a priori soddisfa le assunzioni di indipendenza allora anche la distribuzione a posteriori, essendovi coniugata e perciò appartenente alla stessa famiglia parametrica, le soddisferà. In tal caso è possibile mantenere la distribuzione parametrica in forma chiusa e aggiornarla usando le *statistiche sufficienti*:

- $M[x, x' | pa_i(x)]$ per il parametro $\theta_{x|pa_i(x)}$
- $M[x|pa_i(x)]$ e $T[x|pa_i(x)]$ per il parametro $\mathbf{q}_{x|pa_i(x)}$.

Data una distribuzione sui parametri è possibile usarla per predire il prossimo evento, mediando la sua probabilità sull'insieme dei possibili valori dei parametri. Questo tipo di previsione è equivalente all'utilizzo dei *valori attesi* dei parametri, i quali hanno la stessa forma dei parametri maximum-likelihood ma considerano i *conteggi immaginari* degli iper-parametri:

$$\begin{aligned} \hat{\mathbf{q}}_{x|pa_i(x)} &= \frac{\alpha_{x|pa_i(x)} + M[x|pa_i(x)]}{\tau_{x|pa_i(x)} + T[x|pa_i(x)]} \\ \hat{\theta}_{xx'|pa_i(x)} &= \frac{\alpha_{xx'|pa_i(x)} + M[x, x'|pa_i(x)]}{\alpha_{x|pa_i(x)} + M[x|pa_i(x)]}. \end{aligned} \tag{1.21}$$

Si osservi che questi parametri sono, teoricamente, validi solo per predire una singola transizione, dopo la quale la distribuzione dei parametri andrebbe aggiornata di conseguenza. Tuttavia, com'è spesso fatto in situazioni di questo tipo, si approssima la stima bayesiana congelando i parametri ai succitati *valori attesi*, usandoli per la predizione di una intera traiettoria (i. e. la distribuzione dei parametri non viene aggiornata ad ogni singola transizione).

2 CLASSIFICAZIONE

La classificazione è un argomento centrale nei campi di ricerca relativi all'apprendimento automatico (anche detto *machine learning*) e l'analisi dei dati. In generale, essa consiste nel processo di assegnare una *classe* (i. e. un'etichetta) a delle istanze descritte da un insieme di attributi. Si parla di *classificazione supervisionata* quando è necessario indurre un classificatore a partire da un insieme di dati composto da istanze già etichettate e utilizzare tale classificatore per classificare nuove istanze di dati.

In questo capitolo viene quindi introdotta una classe di modelli, che prende il nome di Continuos time Bayesian Network classifier (CTBNC), il cui scopo è la *classificazione supervisionata* di traiettorie multivariate di variabili discrete a *tempo continuo*. Si descrivono due istanze di tale classe: i classificatori Continuos time Naïve Bayes (CTNB) e i classificatori Continuos time tree augumented Naïve Bayes (CTTANB).

Mentre nella sezione 2.2 si affronta il processo di *apprendimento* in caso di *dati completi* dei CTBNC, nella sezione 2.3, si presenta un algoritmo di *inferenza esatta* per la classe dei CTBNC.

2.1 MODELLO

Al fine di risolvere il succitato problema della classificazione sono stati proposti numerosi approcci. Ad esempio naïve Bayes classifier, un classificatore semplice ma robusto proposto da Duda e Hart (1973); rivelatosi essere uno fra i classificatori più performanti (Langley *et al.*, 1992). Esso apprende dai dati la probabilità condizionale di ogni attributo A_i data la classe C. La classificazione di nuove istanze dei dati è effettuata applicando la *regola di Bayes* al fine di calcolare la probabilità della classe C data l'istanziazione di $A_i, \ldots, A_N$ e scegliendo quella con la maggiore probabilità a posteriori. Questo calcolo è reso possibile da un'assunzione forte: tutti gli attributi A_i sono *condizionalmente indipendenti* (si veda la definizione 2) tra di loro data evidenza sulla classe C.

Poiché tale assunzione è chiaramente irreale, Friedman *et al.* (1997) ha investigato come migliorare ulteriormente le prestazioni del naïve Bayes classifier evitando assunzioni di indipendenza non giustificate dai dati. A tal fine Friedman *et al.* (1997), generalizzando il naïve Bayes classifier, ha proposto una classe di modelli di *classificazione supervisionata*, chiamata Bayesian Network classifier (BNC) (di cui fa

parte il Tree Augumented Naïve Bayes (TAN) classifier, ad esempio) che ereditano dalla teoria delle Bayesian Network (si rimanda alla definizione 1 per maggiori dettagli) una rappresentazione fattorizzata delle distribuzioni di probabilità dei nodi attributo e rappresentano esplicitamente le indipendenze condizionali fra essi.

Seguendo le stesse motivazioni, in Stella e Amer (2012) viene formalizzata una classe di modelli di *classificazione supervisionata*, chiamati Continuos time Bayesian Network classifier (CTBNC), derivata dalle CTBN (si veda definizione 15).

Di seguito si definiscono quindi i Continuos time Bayesian Network classifier e due istanze di classificatori appartenenti a tale classe: il Continuos time Naïve Bayes classifier (CTNBC) e il Continuos time tree augumented Naïve Bayes classifier (CTTANBC).

Un Continuos time Bayesian Network classifier estende una CTBN tramite l'aggiunta di un nodo associato alla variabile classe Y. Si ricorda, dalla definizione 15, che una CTBN rappresenta l'evoluzione nel tempo continuo di una variabile di processo $\mathbf{X}$ (i. e. insieme composto da N processi di Markov, si veda la definizione 11).

Di seguito si dà la definizione di questa nuova classe di modelli di *classificazione supervisionata*.

Definizione 17 (Continuos time Bayesian Network classifier)**.** Un Continuos time Bayesian Network classifier (CTBNC) è composto da una coppia $\mathcal{C} = (\mathcal{N}, \mathbf{P}(Y))$ dove:

- $\mathcal{N}$ è una CTBN con nodi attributo $X_1, X_2, \ldots, X_N$
- Y è il nodo classe con valori $val(Y) = \{y_1, \ldots, y_K\}$ e probabilità marginale $\mathbf{P}(Y)$.

E inoltre il grafo su $\mathcal{N}$ (i. e. il grafo $\mathcal{G}$, si veda la definizione 15) rispetta le seguenti condizioni:

- $\mathcal{G}$ è un grafo connesso[11]
- $Pa(Y) = \{\}$, i. e. la variabile casuale Y è associata a un nodo radice[12]
- il nodo Y è indipendente dal tempo ed è specificato solo ed esclusivamente dalla sua probabilità marginale $\mathbf{P}(Y)$.

A supporto della definizione 17, la figura figure 2.1 nella pagina seguente fornisce un'istanza di CTBNC composta dai nodi attributi X_1, X_2, X_3, X_4, X_5 e dal nodo classe Y (nodo radice). Si osservi come tale istanza contenga dei cicli, uno riguardante i nodi X_2, X_4, X_5, X_3 e l'altro riguardante i nodi X_1, X_3. Si fa notare che gli archi della rete $\mathcal{N}$ rappresentano le dipendenze causali nel tempo.

11 Il grafo $\mathcal{G} = (V, E)$ è detto *connesso* se $\forall (u, v) \in V$ esiste un cammino che collega u a v.

12 In un grafo un nodo è detto *radice* qualora esso non abbia alcun genitore.

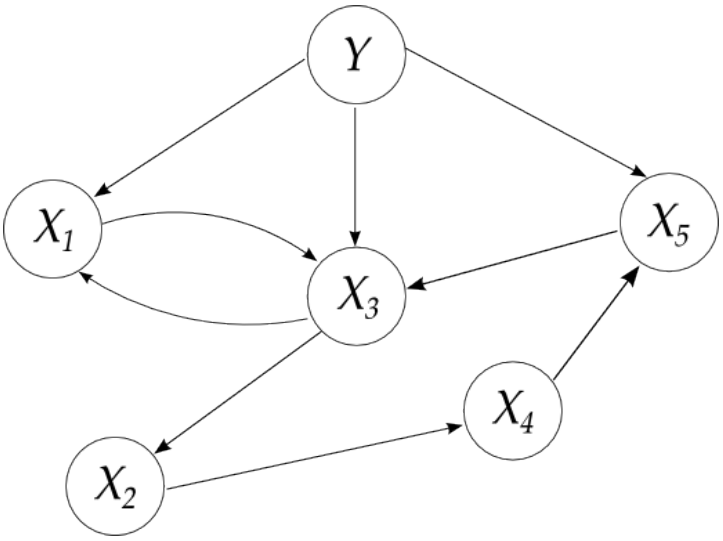

Figura 2.1: Un esempio di Continuos time Bayesian Network classifier (CTBNC) con cinque nodi attributo, $X_1, \ldots, X_5$, e un nodo classe, Y.

Parallelamente a quanto fatto in Langley *et al.* (1992), si presentano ora due istanze particolari di Continuos time Bayesian Network classifier.

Definizione 18 (Continuos time Naïve Bayes classifier)**.** Un Continuos time Naïve Bayes classifier (CTNBC) è un Continuos time Bayesian Network classifier $\mathcal{C} = (\mathcal{N}, \mathbf{P}(Y))$ caratterizzato dal fatto che ogni nodo attributo ha un solo genitore, il nodo classe Y. Risulta quindi che:

$$\mathrm{Pa}(X_i) = \{Y\} \quad \forall\, X_i \in \mathcal{G}.$$

Come mostrato dalla figura figure 2.2 in questa pagina, un CTNBC possiede un nodo radice, associato alla variabile casuale Y, che è l'unico genitore di tutti i restanti nodi X_i (con $i = 1, 2, \ldots, N$) che lo compongono. Si osservi come la rete di un CTNBC rappresenti l'assunzione di *indipendenza condizionale* di ogni nodo attributo dagli altri, data evidenza sulla variabile classe Y.

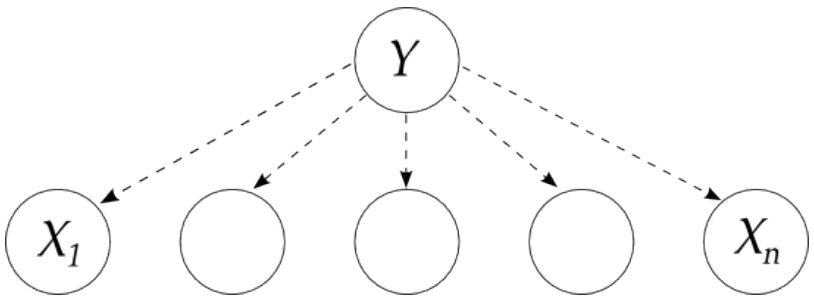

Figura 2.2: Un Continuos time Naïve Bayes classifier (CTNBC).

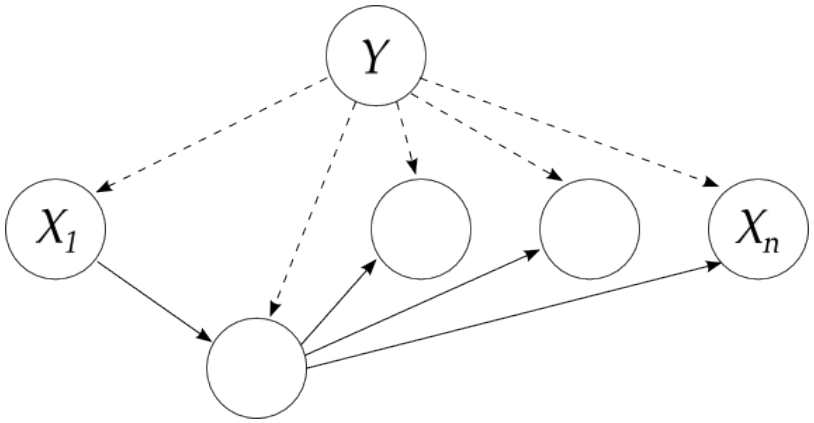

Figura 2.3: Un Continuos time tree augumented Naïve Bayes classifier (CTTANBC): qualora la variabile classe Y venga rimossa, le variabili rimanenti formano un albero.

Definizione 19 (Continuos time tree augumented Naïve Bayes classifier)**.** Un Continuos time tree augumented Naïve Bayes classifier (CTTANBC) è un Continuos time Bayesian Network classifier $\mathcal{C} = (\mathcal{N}, \mathbf{P}(Y))$ che rispetta i seguenti vincoli:

- $Y \in Pa(X_i)$ con $i = 1, 2, \ldots, N$
- i nodi attributo X_i, $i = 1, 2, \ldots, N$, formano un albero:

$$\exists\, j \;:\quad |Pa(X_j)| = 1 \quad \text{mentre per} \quad i \neq j \;:\quad |Pa(X_i)| = 2.$$

Come mostrato dalla figura 2.3 un classificatore CTTANB è un estensione del classificatore CTNB: tutti i nodi attributo della rete $\mathcal{N}$ sono vincolati ad avere come genitore, oltre al nodo radice, al massimo un altro nodo attributo. Ciò comporta che tutti i nodi attributo facciano parte del *Markov blanket*[4] del nodo radice associato con la variabile classe Y.

2.2 APPRENDIMENTO

In questa sezione si affronta il problema dell'apprendimento (da *dati completi*) dei CTBNC.

Per definizione (si veda 17) i CTBNC sono basati sul modello delle CTBN, rappresentano perciò un insieme di modelli di probabilità locali (relativi alle variabili casuali) esprimibili in termini di statistiche sufficienti (per maggiori dettagli relativi a questo aspetto si rimanda alla sezione 1.4). Ne deriva che il problema dell'apprendimento di un classificatore CTBN si riduce alla computazione delle statistiche sufficienti dei suoi nodi attributo, da cui è successivamente possibile (nonché semplice) stimare i parametri (argomento trattato in dettaglio nella sottosezione 1.4.3) delle distribuzioni di probabilità codificate dalle matrici di intensità condizionali (CIM).

Di conseguenza, per l'apprendimento di un CTNBC è richiesto uno sforzo computazionale minimo. Per l'apprendimento di un CTTANBC, poiché questo modello prevede archi anche fra i nodi attributo, è invece richiesto uno sforzo computazionale leggermente maggiore (Stella e Amer, 2012).

Si presenta di seguito l'algoritmo 2.1 relativo all'apprendimento di un classificatore CTNB (definizione 18).

Esso richiede in input un *dataset completo* $\mathcal{D} = \{\delta_1, \ldots, \delta_h\}$, il corrispettivo insieme delle classi $\{y_1, y_2, \ldots, y_h\}$, con $y_i \in val(Y)$, e il grafo $\mathcal{G}$ di una CTBN $\mathcal{N}$ (rispettivamente chiamati `data`, `classes` e `graph` nella firma della funzione `ctnbclearn`).

Per completezza si osservi (in base alla definizione 16) che ogni δ_i (con $i = 1, \ldots, h$) è un flusso di evidenze $(\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^{J_i})$ (a tal riguardo si veda la definizione 14).

Il risultato dell'applicazione dell'algoritmo 2.1 è un Continuos time Naïve Bayes classifier $\mathcal{C} = (\mathcal{N}, \mathbf{P}(Y))$.

Algoritmo 2.1: Apprendimento di un classificatore CTNB

```
1  function ctnbclearn(data, classes, graph) {
2      var h = len(data)
3      var klass = unique(classes)
4      var nk = len(class)
5      var priors[nk]
6      for (i in index(data)) {
7          var k = index(classes[i])
8          priors[k] = priors[k] + (1 / h)
9      }
10     var m(), t()
11     for (i in index(data)) {
12         var y = classes[i]
13         var j = 1
14         while (t_j ⩽ T_i) {
15             for (n in index(graph.nodes)) {
16                 m(x_n^j, x_n^{j+1}, y) = m(x_n^j, x_n^{j+1}, y) + 1
17                 t(x_n^j, y) = t(x_n^j, y) + (t_j − t_{j−1})
18             }
19             j = j + 1
20         }
21     }
22     var q(), thet()
23     foreach (y in klass) {
24         for (n in index(graph.nodes)) {
25             for (x_n in val(X_n)) {
26                 var mm(x_n, y) = ∑_{x'_n ≠ x_n} m(x_n, x'_n, y)
27                 q(x_n, y) = mm(x_n, y) / t(x_n, y)
28                 thet(x_n, y) = m(x_n, x'_n, y) / mm(x_n, y)
```

```
29              }
30          }
31      }
32      var ctbn = new ctbn(graph, q, thet)
33      return (priors, ctbn)
34  }
```

L'algoritmo (2.1) di apprendimento appena presentato consiste nella stima delle matrici di intensità condizionali (CIM) di ogni variabile casuale di $\mathcal{N}$ per ogni classe $y_i \in Y$. Più in dettaglio, esso è composto da tre fasi consecutive:

1. da linea 2 a linea 9 viene calcolata la *probabilità a priori* della variabile classe Y in base alla frequenza di ogni sua istanziazione $y_i \in Y$ in $\mathcal{D}$

2. da linea 10 a linea 21 vengono calcolate le *statistiche sufficienti* di ogni nodo attributo X_i, con $i = 1, 2, \ldots, N$, sull'insieme di dati di apprendimento (anche detto *training set*) $\mathcal{D}$

3. da linea 22 a linea 31 vengono infine stimati, a partire dalle statistiche sufficienti, i *parametri maximum-likelihood* (MLE).

Si osservi che, poiché il processo di apprendimento è eseguito su un classificatore CTNB, l'algoritmo condiziona sia il calcolo delle statistiche sufficienti (i. e. variabili `t` e `m`) che la stima dei parametri (i. e. variabili `q` e `thet`) di ogni nodo attributo X_i solo ed esclusivamente al valore della variabile classe Y (i. e. variabile `y`). Questa semplificazione è dovuta al vincolo che caratterizza i classificatori CTNB: ogni nodo attributo X_i ha un solo genitore, il nodo associato alla variabile classe Y (si veda la definizione 18).

Come anticipato, al costo di un leggero incremento di complessità computazionale, è possibile estendere l'algoritmo 2.1 al fine di creare un algoritmo di apprendimento generale che apprenda un qualsiasi classificatore CTBN. Affinché tale obiettivo sia raggiunto è necessario rimuovere il succitato vincolo sull'insieme dei genitori di ogni nodo attributo. Mentre il calcolo della probabilità a priori della variabile classe Y non varia, il calcolo delle statistiche sufficienti e la stima dei parametri, invece, necessitano di tale generalizzazione.

Nello specifico:

1. il calcolo delle statistiche sufficienti di ogni nodo attributo X_i va condizionato all'istanziazione attuale (i. e. al tempo `j`) del suo insieme di nodi genitori $Pa(X_i)$; perciò a linea 15 dell'algoritmo 2.2 si prende in considerazione tale valore (i. e. variabile `p`)

2. la stima dei parametri maximum-likelihood (MLE) di ogni noto attributo X_i va eseguita in base a ogni istanziazione del suo insieme di nodi genitori $Pa(X_i)$; perciò a linea 25 si itera in

base a ogni valore (i.e. variabile p) assunto da $Pa(X_i)$ (i.e. $val(Pa(X_i))$) mentre l'iterazione per classe non è più coerente e di conseguenza rimossa.

Si riporta di seguito l'algoritmo 2.2, il quale include le succitate modifiche finalizzate alla creazione di un algoritmo di apprendimento generale per i CTBNC.

Algoritmo 2.2: Apprendimento di un classificatore CTBN

```
function learn(data, classes, graph) {
    var h = len(data)
    var klass = unique(classes)
    var nk = len(class)
    var priors[nk]
    for (i in index(data)) {
        var k = index(classes[i])
        priors[k] = priors[k] + (1 / h)
    }
    var m(), t()
    foreach (i in index(data)) {
        var j = 1
        while (t_j ⩽ T_i) {
            foreach (n in index(graph.nodes)) {
                var p = val^j(Pa(X_n))
                m(x_n^j, x_n^{j+1}, p) = m(x_n^j, x_n^{j+1}, p) + 1
                t(x_n^j, p) = t(x_n^j, p) + (t_j − t_{j−1})
            }
            j = j + 1
        }
    }
    var q(), thet()
    foreach (n in index(graph.nodes)) {
        for (x_n in val(X_n)) {
            foreach (p in val(Pa(X_n))) {
                var mm(x_n, p) = ∑_{x'_n ≠ x_n} m(x_n, x'_n, p)
                q(x_n, p) = mm(x_n, p) / t(x_n, p)
                thet(x_n, p) = m(x_n, x'_n, p) / mm(x_n, p)
            }
        }
    }
    var ctbn = new ctbn(graph, q, thet)
    return (priors, ctbn)
}
```

2.3 INFERENZA

In questa sezione si affronta il problema della *classificazione* di un *flusso di evidenze completamente osservato*, indicato con $(\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^J)$ (si veda la definizione 14), rispetto a un classificatore CTBN (CTBNC). Si osservi che, in tale situazione (i. e. *dati completi*), l'unica variabile casuale non osservata è la variabile classe, perciò è possibile sfruttare le relazioni di indipendenza fra variabili casuali così come si fa per le Bayesian Network. L'argomento di questa sezione è quindi il processo di *classificazione supervisionata* (i. e. è necessario un CTBNC appreso da un *training set*), di cui si presentano in primis le basi teoriche e successivamente l'implementazione algoritmica che ne consegue.

Il processo di classificazione di un flusso di evidenze è effettuato in base alla regola *maximum a posteriori*[13] (MAP) (si veda Stella e Amer, 2012): un flusso di evidenze completamente osservato viene classificato assegnandogli la classe la cui probabilità a posteriori (rispetto al flusso di evidenze stesso) è massima. A tale scopo è necessario calcolare la probabilità a posteriori della variabile classe Y del CTBNC, rispetto al flusso di evidenze in input, per tutti i suoi possibili stati (i. e. classi, o etichette).

Il classificatore CTBN classifica quindi il flusso di evidenze massimizzando la seguente probabilità a posteriori, ricavata applicando la *regola di Bayes*:

$$P(Y \mid (\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^J)) = \frac{P((\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^J) \mid Y)\, P(Y)}{P((\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^J))}. \tag{2.1}$$

Si specifica di seguito la semantica dei componenti dell'equazione 2.1:

- la probabilità marginale associata alla variabile classe Y

$$P(Y)$$

- la probabilità del flusso di evidenze

$$P((\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^J))$$

- la likelihood[9] del flusso di evidenze dato il valore della variabile classe, a cui ci si riferisce nel prosieguo usando l'espressione *"likelihood temporale"*

$$P((\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^J) \mid Y).$$

13 La stima della probabilità maximum a posteriori (MAP) è una moda della distribuzione a posteriori che può essere usata per ottenere una stima puntuale di una quantità inosservata sulla base di dati empirici. Può essere vista come una regolarizzazione della stima maximum-likelihood (MLE) poiché è strettamente correlata ad essa; vi differisce perché impiega un obiettivo di massimizzazione incrementato che incorpora una distribuzione a priori sopra la quantità che si vuole stimare.

La probabilità del flusso di evidenze, similmente a quanto accade per i BNC (Friedman *et al.*, 1997), non è richiesta per l'implementazione della regola MAP, perciò è possibile ometterla e riscrivere la precedente equazione sotto forma di relazione proporzionale:

$$P(Y \mid (\mathbf{x}^1, \mathbf{x}^2, \dots, \mathbf{x}^J)) \propto P((\mathbf{x}^1, \mathbf{x}^2, \dots, \mathbf{x}^J) \mid Y)\, P(Y). \tag{2.2}$$

Il primo termine dell'equazione 2.2, cioè la *likelihood temporale*, è invece fondamentale per la classificazione tramite regola MAP ed è possibile riformularlo nel seguente modo:

$$P((\mathbf{x}^1, \mathbf{x}^2, \dots, \mathbf{x}^J) \mid Y) = \prod_{j=1}^{J} P(\mathbf{x}^j \mid Y)\, P(\mathbf{x}^{j+1} \mid \mathbf{x}^j, Y), \tag{2.3}$$

dove:

- $P(\mathbf{x}^j \mid Y)$ rappresenta la probabilità che il *vettore aleatorio*[14] $\mathbf{X}$ resti nello stato $\mathbf{x}^j$ durante l'intervallo temporale $[t_{j-1}, t_j)$ data evidenza sulla variabile classe Y
- $P(\mathbf{x}^{j+1} \mid \mathbf{x}^j, Y)$ rappresenta la probabilità che in $\mathbf{X}$ si verifichi una transizione da $\mathbf{x}^j$ a $\mathbf{x}^{j+1}$ all'istante di tempo t_j data evidenza sulla variabile classe Y.

Inoltre, al fine di assicurare la consistenza dell'equazione 2.3 si assume che $P(\mathbf{x}^{J+1} \mid \mathbf{x}^J, Y) = 1$.

Il passo successivo consiste nel calcolo dei due termini da cui è composta l'equazione 2.3. A tal fine si utilizzano le distribuzioni di probabilità locali associate ad ogni nodo del CTBN su cui è costruito il classificatore. Come già descritto nella sottosezione 1.1.2, tali modelli di probabilità sono espressi tramite le matrici di intensità condizionali e quindi tramite i parametri $\mathbf{q}$ e θ.

In tale contesto è quindi possibile calcolare il termine $P(\mathbf{x}^j \mid Y)$ come segue:

$$P(\mathbf{x}^j \mid Y) = \prod_{n=1}^{N} \exp\Big(-\mathbf{q}_{x_n^j}^{pa_j(x_n)} (t_j - t_{j-1}) \Big), \tag{2.4}$$

dove $\mathbf{q}_{x_n^j}^{pa_j(x_n)}$ è il valore del parametro della *distribuzione esponenziale* quando la variabile casuale X_n è nello stato x_n durante il j-esimo intervallo temporale $[t_{j-1}, t_j)$ e contemporaneamente l'istanziazione dei genitori di X_n è $pa_j(x_n)$.

Ugualmente, il termine $P(\mathbf{x}^{j+1} \mid \mathbf{x}^j, Y)$ è così calcolabile:

$$P(\mathbf{x}^{j+1} \mid \mathbf{x}^j, Y) = \prod_{n=1}^{N} P(x_n^{j+1} \mid x_n^j, Y), \tag{2.5}$$

14 Un *vettore aleatorio* $\mathbf{X} = (X_1, X_2, \dots, X_n)$ è una n-*upla* (composta da n variabili casuali) i cui elementi sono dati da numeri aleatori.

dove

$$P(x_n^{j+1} \mid x_n^j, Y) = \begin{cases} \mathbf{q}^{pa_j(x)}_{x_n^j x_n^{j+1}}, & \text{se } x_n^j \neq x_n^{j+1} \\ 1, & \text{altrimenti} \end{cases}. \tag{2.6}$$

Il termine $P(x_n^{j+1} \mid x_n^j, Y)$ rappresenta la probabilità che nel vettore aleatorio $\mathbf{X}$ si verifichi una transizione dallo stato $\mathbf{x}^j$ allo $\mathbf{x}^{j+1}$, dato il valore della variabile classe Y.

Poiché il modello CTBN implica che, ad ogni istante t_j, solo un componente X_n del vettore aleatorio $\mathbf{X}$ può essere soggetto a transizione, allora $P(x_n^{j+1} \mid x_n^j, Y)$ rappresenta la probabilità che la variabile casuale X_n effettui una transizione da x_n^j a x_n^{j+1} mentre tutti gli altri componenti del vettore aleatorio $\mathbf{X}$ (i. e. X_i con $i \neq n$) non cambiano il proprio stato.

Perciò, come specificato dall'equazione 2.6, nel caso in cui avvenga un cambio di stato in X_n, il termine $P(x_n^{j+1} \mid x_n^j, Y)$ equivale alla quantità $\mathbf{q}^{pa_j(x)}_{x_n^j x_n^{j+1}}$, ricavata dalla relazione fra i parametri (si veda il teorema 1.2). Tale quantità rappresenta il parametro associato alla transizione da x_n^j, stato in cui la variabile casuale X_n si trovava durante il j-esimo intervallo temporale $[t_{j-1}, t_j)$, a x_n^{j+1}, stato in cui X_n si troverà durante il successivo intervallo temporale $[t_j, t_{j+1})$; data l'istanziazione $pa_j(x_n)$ dei genitori di X_n durante il j-esimo intervallo temporale.

Combinando l'equazione 2.4 e l'equazione 2.5 si ottiene:

$$P(\mathbf{x}^j \mid Y)\, P(\mathbf{x}^{j+1} \mid \mathbf{x}^j, Y) = \prod_{n=1}^{N} \exp\Big(-\mathbf{q}^{pa_j(x_n)}_{x_n^j} (t_j - t_{j-1}) \Big)\, P(x_n^{j+1} \mid x_n^j, Y). \tag{2.7}$$

Utilizzando l'equazione 2.7 appena ricavata è possibile riformulare l'equazione della *likelihood temporale* (2.3) nel seguente modo:

$$P((\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^J) \mid Y) = \prod_{j=1}^{J} \prod_{n=1}^{N} \exp\Big(-\mathbf{q}^{pa_j(x_n)}_{x_n^j} (t_j - t_{j-1}) \Big)\, P(x_n^{j+1} \mid x_n^j, Y). \tag{2.8}$$

Sostituendo infine l'equazione appena scritta (2.8) nell'equazione 2.2 si formula definitivamente la probabilità a posteriori della variabile classe Y dato un flusso di evidenze:

$$P(Y \mid (\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^J)) \propto P(Y) \cdot \prod_{j=1}^{J} \prod_{n=1}^{N} \Bigg[\exp\Big(-\mathbf{q}^{pa_j(x_n)}_{x_n^j} (t_j - t_{j-1}) \Big) \cdot \cdot P(x_n^{j+1} \mid x_n^j, Y) \Bigg]. \tag{2.9}$$

Di conseguenza, dato un classificatore CTBN $\mathcal{C} = \{\mathcal{N}, P(Y)\}$ e un flusso di evidenze completamente osservato $(\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^J)$, la regola MAP seleziona la classe $y^* \in val(Y)$ massimizzando l'equazione 2.9:

$$y^* = \arg\max_{y \in val(Y)} P(Y) \prod_{j=1}^{J} \prod_{n=1}^{N} \exp\Big(-\mathbf{q}^{pa_j(x_n)}_{x_n^j} (t_j - t_{j-1}) \Big)\, P(x_n^{j+1} \mid x_n^j, Y) \tag{2.10}$$

Si presenta di seguito l'algoritmo per l'*inferenza esatta*) di un flusso di evidenze completamente osservato rispetto a un classificatore CTBN.

Tuttavia si osservi che tale algoritmo rappresenta le probabilità come *log-probabilità*[15] per motivi computazionali. Ciò significa che l'algoritmo 2.3 implementa la probabilità a posteriori della classe dato un flusso di evidenze (equazione 2.9) come segue:

$$\ell_{P(Y|\ldots)} = \log(P(Y)) + \sum_{j=1}^{J} \sum_{n=1}^{N} -\mathbf{q}_{x_n^j}^{pa_j(x_n)} (t_j - t_{j-1}) + \log(P(x_n^{j+1} \mid x_n^j, Y)). \tag{2.11}$$

A tal proposito si noti che, nel caso in cui non avvenga alcun cambio di stato durante un determinato istante di tempo t_j, la quantità $P(x_n^{j+1} \mid x_n^j, Y)$ in 2.11 sarà pari a 1 (si veda l'equazione 2.6), il cui logaritmo è pari a 0. Ciò si riflette nella struttura di controllo condizionale alla linea 11.

Algoritmo 2.3: Inferenza su un classificatore CTBN

```
1  function infer(ctbnc, timeseg, stream) {
2      var priors = ctbnc.priors
3      var logp[len(priors)]
4      for (k in index(priors)) {
5          logp[k] = log(priors[k])
6      }
7      for (k in index(priors)) {
8          for (j in index(timeseg)) {
9              for (n in index(ctbnc.graph.nodes)) {
10                 logp[k] = logp[k] - q^{pa_j(x_n)}_{x_n^j} * timeseg[j]
11                 if (x_{n_j} != x_{n_{j+1}}) {
12                     logp[k] = logp[k] + log(q^{pa_j(x)}_{x_n^j x_n^{j+1}})
13                 }
14             }
15         }
16     }
17     return which(max(logp))
18 }
```

L'algoritmo 2.3 di inferenza esatta appena presentato è composto principalmente da due fasi corrispondenti ai due termini principali dell'equazione 2.11: da linea 2 a linea 6 converte la distribuzione della variabile classe Y del classificatore CTBN in forma logaritmica; da linea 7 a linea 16 aggiorna, incrementandola o decrementandola, il

15 La *log-probabilità* è un modalità di rappresentazione della probabilità che porta con sè alcuni vantaggi computazionali. Ad esempio, l'utilizzo della *log-probabilità* generalmente comporta una maggior velocità dovuta alla trasformazione delle moltiplicazioni, più computazionalmente costose, in addizioni. In informatica è molto comune l'utilizzo della sua variante negativa, la quale codifica un valore di probabilità $x \in [0, 1]$ come $x' = -\log(x) \in \mathbb{R}$.

valore di *log-probabilità* relativo a ogni classe (si veda il ciclo `for` alla linea 7) del classificatore CTBN iterando il flusso di evidenze per ogni segmento temporale cui sono associate le sue istanze (i. e. variabile `timeseg`, che si assume venga fornita in input; si veda il ciclo `for` alla linea 8), infine iterando sui nodi del grafo $\mathcal{N}$ del CTBNC (ciclo `for` a linea 9).

Nota 2.3.1. Si osservi che è possibile implementare l'algoritmo di inferenza anche utilizzando la *parametrizzazione mista* (si veda la definizione 8). In tal caso si evita la computazione delle matrici di intensità condizionali (CIM) ma, in contrasto, è necessario calcolare il termine $\mathbf{q}^{\mathrm{pa}_j(x)}_{x_n^j x_n^{j+1}}$ (a tal proposito si veda l'equazione 1.14) ogni qual volta una variabile casuale del flusso di evidenze di input effettui una transizione (a patto che tale transizione avvenga fra due stati appartenenti allo spazio degli stati della rispettiva variabile casuale associata al nodo del classificatore CTBN).

Il passo finale dell'algoritmo, corrispondente alla linea finale (i. e. linea 17), consiste nella restituzione della classe la cui *log-probabilità* è maggiore di quella delle restanti classi. Tale passo completa perciò l'implementazione dell'equazione 2.10.

Nota 2.3.2. Si noti che l'algoritmo in esame implementa una conversione delle probabilità in forma logaritmica non negativa (i. e. $x \in [0, 1]$ rappresentato come $x' = \log(x)$). Ne consegue che l'applicazione della regola MAP non varia.

Anche l'algoritmo di inferenza esatta presentato in questa sezione è, così come l'algoritmo di apprendimento, computazionalmente efficiente.

3 APPRENDIMENTO STRUTTURALE

Uno dei casi principali che costituisce il problema dell'*apprendimento* di modelli grafico probabilistici è l'apprendimento della struttura incognita sottostante ogni modello.

Il problema dell'*apprendimento strutturale* da *dati completi* di una Continuos time Bayesian Network (CTBN) è quindi l'argomento trattato in questo capitolo.

Generalmente, questo problema può essere informalmente descritto nel seguente modo: dato un training set composto da istanze di un insieme di variabili casuali si trovi un grafo che rappresenti tali dati e le relazioni fra le variabili casuali. Si noti come tale problema possa essere catalogato come una forma di apprendimento non supervisionato; nel senso che il processo di apprendimento non distingue la variabile classe dalle variabili attributo nei dati. L'obiettivo è quindi indurre una struttura (i. e. grafo) che descriva nel miglior modo possibile la distribuzione di probabilità sui dati (i. e. *training set*). Si osservi, inoltre, che questo problema di ottimizzazione è solitamente intrattabile per le Bayesian Network (Chickering, 1994; Chickering *et al.*, 2004), anche qualora si restringa la cardinalità massima dell'insieme dei genitori di ogni nodo.

Per quanto riguarda invece il caso delle CTBN, Nodelman *et al.* (2002) hanno dimostrato che, grazie alla mancanza del vincolo di aciclicità, come già accennato nella sezione 1.3), il problema dell'apprendimento strutturale di una CTBN è significativamente più facile, sia teoricamente che in pratica, rispetto all'apprendimento strutturale di una Bayesian Network, o di modelli da esse derivanti, e. g. le Dynamic Bayesian Networks (DBN). Inoltre, nel caso si vincoli la procedura di ricerca a strutture con un numero massimo di genitori per nodo, questo problema può essere risolto in tempo polinomiale.

L'approccio che si presenta in questo capitolo è quindi un approccio basato sul punteggio: si definisce una funzione che computa uno *score bayesiano* finalizzato alla valutazione di ogni struttura rispetto ai dati di addestramento e si usa una tecnica di ricerca euristica (ad esempio, la ricerca *hill climbing*) per cercare nello spazio delle strutture candidate quella che esibisce il maggior punteggio.

Si osservi che l'apprendimento dei parametri (si veda la "Stima bayesiana", sottosezione 1.4.3) è propedeutico per tale obiettivo poiché essi costituiscono la base dello score bayesiano.

3.1 FUNZIONE DI SCORING

Qualsiasi processo di apprendimento strutturale basato su punteggio è costituito da due componenti: una *funzione di scoring* e una procedura di ottimizzazione.

L'obiettivo di questa sezione è quindi presentare una funzione di scoring per l'apprendimento strutturale delle Continuos time Bayesian Network (CTBN). Lo scopo di tale funzione è calcolare il punteggio (i. e. lo *score*) di una struttura relativamente al *training set* $\mathcal{D}$ fornito.

Si definisce lo *score bayesiano* sul grafo $\mathcal{G}$ di una CTBN nel seguente modo:

$$\text{score}_B(\mathcal{G} : \mathcal{D}) = \ln P(\mathcal{D} \mid \mathcal{G}) + \ln P(\mathcal{G}) \tag{3.1}$$

Come mostra l'equazione 3.1 la funzione di scoring utilizza la probabilità a posteriori dell'insieme dei dati di apprendimento (i. e. il training set $\mathcal{D}$) data la struttura candidata (i. e. $\mathcal{G}$), oltre alla probabilità a priori della struttura stessa.

È possibile aumentare in modo significativo l'efficienza dell'algoritmo di ricerca che si affronta nella prossima sezione qualora si facciano determinate assunzioni. Nello specifico, se si assume che la probabilità a priori della struttura, $P(\mathcal{G})$, soddisfi la *modularità della struttura*, ne consegue:

$$P(\mathcal{G}) = \prod_{X_i} P(Pa(X_i) = Pa_{\mathcal{G}}(X_i)). \tag{3.2}$$

Se si assume, inoltre, che la probabilità a priori dei parametri soddisfi la *modularità dei parametri*, allora per ogni due strutture $\mathcal{G}$ e $\mathcal{G}'$ tali che $Pa_{\mathcal{G}}(X) = Pa_{\mathcal{G}'}(X)$ risulta:

$$P(\mathbf{q}_X, \theta_X \mid \mathcal{G}) = P(\mathbf{q}_X, \theta_X \mid \mathcal{G}'). \tag{3.3}$$

Combinando l'assunzione di *indipendenza dei parametri* con l'equazione 3.3 derivante dalla *modularità dei parametri*, si ottiene:

$$P(\mathbf{q}_{\mathcal{G}}, \theta_{\mathcal{G}} \mid \mathcal{G}) = \prod_{X_i} \left[P\left(\mathbf{q}_{X_i \mid Pa(X_i)} \mid Pa(X_i) = Pa_{\mathcal{G}}(X_i)\right) \cdot \right.$$
$$\left. \cdot P\left(\theta_{X_i \mid Pa(X_i)} \mid Pa(X_i) = Pa_{\mathcal{G}}(X_i)\right) \right]. \tag{3.4}$$

Si osservi che, poiché la *penalità del grafo*, corrispondente al termine $P(Pa(X_i) = Pa_{\mathcal{G}}(X_i))$ dell'equazione 3.2, è legata alla dimensione del grafo ma indipendente dalla quantità dei dati, è possibile ignorare il termine $P(\mathcal{G})$ della funzione di scoring (equazione 3.1).

Di conseguenza l'unico termine significativo dell'equazione 3.1 è la *likelihood marginale*, $P(\mathcal{D} \mid \mathcal{G})$. Tale termine, infatti, codifica l'incertezza sui parametri integrando su tutti i possibili valori che essi possono assumere:

$$P(\mathcal{D} \mid \mathcal{G}) = \int_{\mathbf{q}_{\mathcal{G}}, \theta_{\mathcal{G}}} P(\mathcal{D} \mid \mathbf{q}_{\mathcal{G}}, \theta_{\mathcal{G}})\, P(\mathbf{q}_{\mathcal{G}}, \theta_{\mathcal{G}} \mid \mathcal{G})\, d\mathbf{q}_{\mathcal{G}}\, d\theta_{\mathcal{G}}. \tag{3.5}$$

Come per l'equazione 1.8, la likelihood marginale può essere decomposta come un prodotto di likelihood:

$$\begin{aligned} P(\mathcal{D} \mid \mathbf{q}_{\mathcal{G}}, \boldsymbol{\theta}_{\mathcal{G}}) &= \prod_{X_i} L_{X_i}(\mathbf{q}_{X_i \mid Pa(X_i)} : \mathcal{D})\, L_{X_i}(\boldsymbol{\theta}_{X_i \mid Pa(X_i)} : \mathcal{D}) \\ &= \underbrace{\left[\prod_{X_i} L_{X_i}(\mathbf{q}_{X_i \mid Pa(X_i)} : \mathcal{D})\right]}_{L(\mathbf{q}:\mathcal{D})} \underbrace{\left[\prod_{X_i} L_{X_i}(\boldsymbol{\theta}_{X_i \mid Pa(X_i)} : \mathcal{D})\right]}_{L(\boldsymbol{\theta}:\mathcal{D})}. \end{aligned} \tag{3.6}$$

Combinando tale decomposizione con l'*indipendenza dei parametri* si può riformulare la likelihood marginale (equazione 3.5) nel seguente modo:

$$\begin{aligned} P(\mathcal{D} \mid \mathcal{G}) &= \int_{\mathbf{q}_{\mathcal{G}}, \boldsymbol{\theta}_{\mathcal{G}}} L(\mathbf{q}_{\mathcal{G}} : \mathcal{D}) L(\boldsymbol{\theta}_{\mathcal{G}} : \mathcal{D}) P(\mathbf{q}_{\mathcal{G}}) P(\boldsymbol{\theta}_{\mathcal{G}})\, d\mathbf{q}_{\mathcal{G}} d\boldsymbol{\theta}_{\mathcal{G}} \\ &= \underbrace{\left[\int_{\mathbf{q}_{\mathcal{G}}} L(\mathbf{q}_{\mathcal{G}} : \mathcal{D}) P(\mathbf{q}_{\mathcal{G}})\, d\mathbf{q}_{\mathcal{G}}\right]}_{(a)} \cdot \underbrace{\left[\int_{\boldsymbol{\theta}_{\mathcal{G}}} L(\boldsymbol{\theta}_{\mathcal{G}} : \mathcal{D}) P(\boldsymbol{\theta}_{\mathcal{G}})\, d\boldsymbol{\theta}_{\mathcal{G}}\right]}_{(b)}. \end{aligned} \tag{3.7}$$

Ottenuta tale equazione, si affronta di seguito l'analisi e la decomposizione dei due termini che la compongono.

Utilizzando l'assunzione di *indipendenza locale dei parametri*, il termine (a) dell'equazione 3.7 è decomponibile nel seguente modo. Si noti che per brevità si pone $u = pa_i(x)$.

$$\prod_{X_i} \prod_{u} \prod_{x} \int_0^{\infty} P(\mathbf{q}_{x \mid u}) \cdot L_{X_i}(\mathbf{q}_{x \mid u} : \mathcal{D})\, d\mathbf{q}_{x \mid u}. \tag{a}$$

Sostituendo a tale termine la distribuzione a priori coniugata su $\mathbf{q}$ (si veda l'equazione 1.16) e la likelihood delle quantità di tempo trascorse in ogni stato (si veda l'equazione 1.10) si ottiene:

$$\prod_{X_i} \prod_{u} \prod_{x} \int_0^{\infty} \frac{(\tau_{x|u})^{\alpha_{x|u}+1}}{\Gamma(\alpha_{x|u}+1)} (\mathbf{q}_{x|u})^{\alpha_{x|u}} e^{-\mathbf{q}_{x|u}\tau_{x|u}} \cdot (\mathbf{q}_{x|u})^{M[x|u]} e^{-\mathbf{q}_{x|u}T[x|u]} d\mathbf{q}_{x|u}. \tag{a}$$

Si procede semplificando:

$$\prod_{X_i} \prod_{u} \prod_{x} \int_0^{\infty} \frac{(\tau_{x|u})^{\alpha_{x|u}+1} \cdot (\mathbf{q}_{x|u})^{\alpha_{x|u}+M[x|u]}}{\Gamma(\alpha_{x|u}+1) \cdot e^{\mathbf{q}_{x|u}(\tau_{x|u}+T[x|u])}}\, d\mathbf{q}_{x|u}. \tag{a}$$

E infine, risolvendo l'integrale, si ottiene:

$$\prod_{X_i} \underbrace{\prod_{u} \prod_{x} \frac{\Gamma(\alpha_{x|u}+M[x|u]+1)(\tau_{x|u})^{\alpha_{x|u}+1}}{\Gamma(\alpha_{x|u}+1)(\tau_{x|u}+T[x|u])^{\alpha_{x|u}+M[x|u]+1}}}_{MargL^{q}(X_i, Pa_{\mathcal{G}}(X_i):\mathcal{D})}. \tag{a}$$

Relativamente all'analisi del termine (b) dell'equazione 3.7 si osservi che, poiché le distribuzioni sui parametri θ sono di *Dirichlet*, tale operazione è analoga a quella comune per le Bayesian Network.

Ne consegue che il termine (b) si semplifica:

$$\underbrace{\prod_{X_i}\prod_{u}\prod_{x}\frac{\Gamma(\alpha_{x|u})}{\Gamma(\alpha_{x|u}+M[x|u])}\cdot\prod_{x\neq x'}\frac{\Gamma(\alpha_{xx'|u}+M[x,x'|u])}{\Gamma(\alpha_{xx'|u})}}_{MargL^{\theta}(X_i,Pa_{\mathcal{G}}(X_i):\mathcal{D})}. \quad \text{(b)}$$

Quindi si può riformulare la likelihood marginale:

$$P(\mathcal{D}|\mathcal{G})=\prod_{X_i}MargL^{q}(X_i,Pa_{\mathcal{G}}(X_i):\mathcal{D})\cdot MargL^{\theta}(X_i,Pa_{\mathcal{G}}(X_i):\mathcal{D}). \tag{3.8}$$

Al fine di derivare la probabilità a priori della struttura (equazione 3.2) si è già assunto in precedenza che l'ipotesi di modularità della struttura sussista. Perciò, sfruttando tale assunzione, l'equazione (3.2) della probabilità a priori della struttura, e l'equazione 3.8) della likelihood marginale:

$$\begin{aligned}score_B(\mathcal{G}:\mathcal{D})=\sum_{X_i}\Big[&\ln P(Pa(X_i)=Pa_{\mathcal{G}}(X_i))+\\&+\ln MargL^{q}(X_i,Pa_{\mathcal{G}}(X_i):\mathcal{D})+\\&+\ln MargL^{\theta}(X_i,Pa_{\mathcal{G}}(X_i):\mathcal{D})\Big]=\\=\sum_{X_i}&famscore_{\mathcal{B}}(X_i,Pa_{\mathcal{G}}(X_i):\mathcal{D}).\end{aligned} \tag{3.9}$$

Si è quindi definita la funzione di scoring come una somma di score bayesiani, $famscore_{\mathcal{B}}(X_i,Pa_{\mathcal{G}}(X_i):\mathcal{D})$, relativi ai nodi del grafo $\mathcal{G}$. Ognuno di tali score bayesiani misura la qualità di $Pa_{\mathcal{G}}(X_i)$ come insieme dei nodi genitori di X_i, dato l'insieme dei dati di apprendimento $\mathcal{D}$.

3.2 RICERCA DELLA STRUTTURA

In questa sezione si affronta il secondo componente del processo di apprendimento strutturale: l'utilizzo di una *procedura di ottimizzazione* finalizzata alla ricerca di una struttura che massimizzi lo *score bayesiano*.

Chickering (1994) ha mostrato come il problema di apprendere la struttura ottimale di una Bayesian Network, detto problema *k-learn*, dove k è il numero massimo di genitori per ogni variabile casuale, sia un problema *NP-arduo* anche qualora si imponga $k = 2$. La ragione di tale complessità è dovuta al vincolo di aciclicità delle BN (i. e. il

grafo di una BN, come da definizione 1, deve essere un DAG[1]): non è perciò possibile determinare l'insieme ottimale dei genitori di ogni nodo di una BN individualmente; poiché la scelta di un insieme di genitori per un nodo restringe la possibilità di scelta relativa ai nodi restanti.

Come già accennato ed intuibile dalla composizione dello score bayesiano (si veda la funzione $famscore_{\mathcal{B}}$, equazione 3.9), la ricerca della struttura ottimale di un modello Continuos time Bayesian Network (CTBN) è invece notevolmente più semplice rispetto a quello relativo alle BN (o alle DBN). La motivazione di tale vantaggio risiede nel fatto che, poiché gli archi fra i nodi del grado $\mathcal{G}$ di una CTBN rappresentano l'effetto del valore attuale della variabile casuale padre sul valore successivo della variabile casuale figlia, non esiste un vincolo di aciclicità ed è di conseguenza possibile ottimizzare l'insieme dei nodi genitori di un qualsiasi nodo separatamente dagli altri.

Inoltre, qualora si restringa il massimo numero di genitori a un valore k, per ogni variabile casuale X_i di una CTBN, con $i = 1, \dots, N$, si può semplicemente enumerare ogni suo possibile insieme di nodi genitori $Pa(X_i)$ tale che $|Pa(X_i)| \leqslant k$ e calcolare il rispettivo punteggio $famscore_{\mathcal{B}}(X_i, Pa(X_i); \mathcal{D})$. Quindi scegliere come insieme dei nodi genitori di X_i quello con punteggio massimo.

Si osservi che, fissato k, questa procedura di ricerca è polinomiale rispetto a N. Si definisce perciò il seguente teorema (Nodelman *et al.*, 2002).

Teorema 3.1 (Problema k-learn)**.** *Il problema* ***k-learn*** *per le Continuos time Bayesian Network, fissato* k*, può essere risolto in tempo polinomiale rispetto al numero di variabili casuali* N *e alla dimensione dell'insieme di dati* $\mathcal{D}$*.*

Si osservi che, fissando k a priori, non è necessario enumerare esaustivamente tutti i possibili insiemi di nodi genitori di ogni nodo di una CTBN. Di conseguenza è possibile utilizzare un algoritmo di ricerca euristica di tipo *greedy* (i. e. goloso) per esplorare lo spazio di ricerca. Nel seguito si presenta perciò l'algoritmo utilizzato, il cui nome è *hill climbing*.

3.2.1 Hill Climbing

...

4 PACKAGE R

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

4.1 ANALISI

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

4.2 PACKAGE CTBN

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus.

Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

5 CREAZIONE DI DATASET RELATIVI AL TRAFFICO

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

5.1 TSIS

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

5.1.1 Descrizione

...

5.1.2 API

...

5.2 ESTENSIONE

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

5.2.1 Analisi

...

5.2.2 Sensors DLL

...

5.3 APPLICATIVI DI SUPPORTO

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

6 ESPERIMENTI NUMERICI

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

6.1 DATASET 1

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

6.1.1 Modello TSIS

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus

eu enim. Vestibulum pellentesque felis eu massa.

6.1.2 Risultati

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

6.2 DATASET 2

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.

6.2.1 Modello TSIS

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

6.2.2 Risultati

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellente-

sque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

7 CONCLUSIONI

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

A GUIDE ALL'USO

A.1 UTILIZZO DEL PACKAGE CTBN

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

A.1.1 Caricamento del dataset

A.1.2 Calcolo delle sufficient statistics

A.1.3 Calcolo dei parametri

A.1.4 Calcolo delle CIM

A.1.5 Apprendimento

A.1.6 Classificazione

A.1.7 Apprendimento strutturale

A.1.8 Cross-validation

A.2 CREAZIONE DI DATASET

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt ur-

na. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

A.2.1 Sensors DLL

...

Installazione

...

Guida all'uso

...

A.2.2 Applicativi di supporto

...

ACRONIMI

BN	Bayesian Network	1
BNC	Bayesian Network classifier	21
CIM	Conditional Intensity Matrix	10
CPD	Conditional Probability Distribution	2
CPT	Conditional Probability Table	2
CTBN	Continuos time Bayesian Network	1
CTBNC	Continuos time Bayesian Network classifier	21
CTNB	Continuos time Naïve Bayes	21
CTNBC	Continuos time Naïve Bayes classifier	22
CTTANB	Continuos time tree augumented Naïve Bayes	21
CTTANBC	Continuos time tree augumented Naïve Bayes classifier	22
DAG	Directed acyclic graph	1
DBN	Dynamic Bayesian Networks	33
EM	Expectation Maximization	4
IC	Inductive Causation	6
IM	Intensity Matrix	8
MAP	Maximum a posteriori	28
MCMC	Markov Chain Monte Carlo	5
MLE	Maximum Likelihood Estimation	18
PV	Process Variable	12
TAN	Tree Augumented Naïve Bayes	22

INDICE ANALITICO

BIBLIOGRAFIA

Chickering, David Maxwell

1994 *Learning Bayesian networks is NP-hard*, rapp. tecn., Microsoft Research. (Citato alle p. 33, 36.)

2013 «A Transformational Characterization of Equivalent Bayesian Network Structures», *CoRR*, p. 87-98, http://arxiv.org/abs/1302.4938. (Citato a p. 6.)

Chickering, David Maxwell, David Heckerman e Christopher Meek

2004 «Large-sample learning of Bayesian networks is NP-hard», *... Journal of Machine Learning ...*, 5, p. 1287-1330, http://dl.acm.org/citation.cfm?id=1044703. (Citato a p. 33.)

Dempster, A P, N M Laird e D B Rubin

1977 «Maximum likelihood from incomplete data via the EM algorithm», *Journal of the Royal Statistical Society Series B Methodological*, Series B, 39, 1, p. 1-38, ISSN: 00359246, DOI: 10.2307/2984875, http://www.jstor.org/stable/2984875. (Citato a p. 5.)

Duda, R. O. e P. E. Hart

1973 *Pattern Classification and Scene Analysis*, John Willey & Sons, New Yotk. (Citato a p. 21.)

Friedman, N, D Geiger e M Goldszmidt

1997 «Bayesian network classifiers», *Machine learning*, 163, p. 131-163, http://link.springer.com/article/10.1023/A\%3A1007465528199. (Citato alle p. 21, 29.)

Geman, Stuart e Donald Geman

1984 «Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images», *IEEE Trans. Pattern Anal. Mach. Intell.*, 6, 6, p. 721-741, ISSN: 0162-8828, DOI: 10.1109/TPAMI.1984.4767596, http://dx.doi.org/10.1109/TPAMI.1984.4767596. (Citato a p. 5.)

Gihman, Iosif I. e Anatolij V. Skorohod

1973 *The theory of stochastic processes II*, New York: Springer-Verlag. (Citato a p. 9.)

Gilks, WR, S Richardson e DJ Spiegelhalter

1996 «Markov chain Monte Carlo in practice». (Citato a p. 5.)

Heckerman, David

1996 «A Tutorial on Learning With Bayesian Networks», *Innovations in Bayesian Networks*, Studies in Computational Intelligence, 1995, November, a cura di Dawn E Holmes e Lakhmi C Jain, p. 33-82, ISSN: 1860949X, DOI: 10.1007/978-3-540-85066-3, http://www.springerlink.com/index/62mv333389016034.pdf. (Citato alle p. 5, 19.)

Heckerman, David, Dan Geiger e David M. Chickering

1995 «Learning Bayesian networks: The combination of knowledge and statistical data», *Machine Learning*, 20, 3 (set. 1995), p. 197-243, ISSN: 0885-6125, DOI: 10.1007/BF00994016, http://link.springer.com/10.1007/BF00994016. (Citato alle p. 6, 19.)

Korb, K.B. e A.E. Nicholson

2011 *Bayesian Artificial Intelligence*, Chapman & Hall / CRC Computer Science and Data Analysis, CRC PressINC, ISBN: 9781439815915. (Citato alle p. 1, 3.)

Langley, Pat, Wayne Iba e Kevin Thompson

1992 «An Analysis of Bayesian Classifiers», in *AAAI*, p. 223-228, ISBN: 0-262-51063-4. (Citato alle p. 21, 23.)

Loève, Michel

1978 *Probability theory. II Edition.* Fourth, Graduate Texts in Mathematics, Vol. 46, Springer-Verlag, New York, p. xvi+413, ISBN: 0-387-90262-7. (Citato alle p. 6, 7.)

MacKay, D. J. C.

1998 «Introduction to Monte Carlo methods», in *Proceedings of the NATO Advanced Study Institute on Learning in graphical models*, Kluwer Academic Publishers, Norwell, MA, USA, p. 175-204, http://dl.acm.org/citation.cfm?id=299068.299077. (Citato a p. 5.)

Nodelman, Uri, CR Shelton e Daphne Koller

2002 «Learning continuous time Bayesian networks», *Proceedings of the Nineteenth ...*, X, arXiv:/arxiv.org/abs/1212.2498 [http:], http://dl.acm.org/citation.cfm?id=2100639. (Citato alle p. 15, 16, 33, 37.)

Nodelman, Uri D.

2007 *Continuos Time Bayesian Networks*, tesi di dott., Stanford University. (Citato alle p. 9, 12, 17-19.)

Norris, James R.

1998 *Markov chains*, Cambridge series in statistical and probabilistic mathematics, Cambridge University Press, p. I-XVI, 1-237, ISBN: 978-0-521-48181-6. (Citato a p. 7.)

Pearl, Judea

1988 *Probabilistic reasoning in intelligent systems: networks of plausible inference*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, ISBN: 0-934613-73-7. (Citato a p. 5.)

Russell, Stuart J. e Peter Norvig

2003 *Artificial Intelligence: A Modern Approach*, Pearson Education, ISBN: 0137903952, http://portal.acm.org/citation.cfm?id=773294. (Citato alle p. 2, 3.)

Shachter, Ross D. e Mark A. Peot

1990 «Simulation Approaches to General Probabilistic Inference on Belief Networks», in *Proceedings of the Fifth Annual Conference on Uncertainty in Artificial Intelligence*, UAI '89, North-Holland Publishing Co., Amsterdam, The Netherlands, p. 221-234, ISBN: 0-444-88738-5, http://dl.acm.org/citation.cfm?id=647232.719570. (Citato a p. 5.)

Steck, Harald and Jaakkola, Tommi S

2002 «On the Dirichlet Prior and Bayesian Regularization», *Advances in Neural Information Processing Systems*, September, p. 713-720. (Citato a p. 19.)

Stella, F e Y Amer

2012 «Continuous time Bayesian network classifiers.», *Journal of biomedical informatics*, 45, 6 (dic. 2012), p. 1108-19, ISSN: 1532-0480, DOI: 10.1016/j.jbi.2012.07.002, http://www.ncbi.nlm.nih.gov/pubmed/22846170. (Citato alle p. 10, 12, 22, 25, 28.)

Verma, Thomas S. e Judea Pearl

1991 «Equivalence and synthesis of causal models», in *Uncertainty in Artificial Intelligence*, North Holland, p. 255-268. (Citato a p. 6.)

DICHIARAZIONE

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque.

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices.

Milano, settembre 2013

Leonardo Di Donato