UNIVERSITÀ DEGLI STUDI DI PADOVA

SEDE AMMINISTRATIVA: UNIVERSITÀ DEGLI STUDI DI PADOVA
DIPARTIMENTO DI SCIENZE STATISTICHE

Dottorato di Ricerca in Statistica - Ciclo XVIII -
Settore Scientifico Disciplinare SECS-S/01

Reti Bayesiane: Approcci per la Selezione del Modello

Coordinatore: Ch.mo Prof. Silvano Bordignon

Supervisore: Ch.mo Prof. Adriana Brogini

Dottoranda: Debora Slanzi

Padova, 31 Dicembre 2005

Ai miei genitori...

Indice

Introduzione

Il termine incertezza è usato per identificare una molteplicità di concetti. L'incertezza può sorgere da una informazione incompleta, da una informazione non disponibile o da un disaccordo tra fonti di informazione. L'incertezza può essere riferita alla variabilità di un fenomeno, ad una quantità o alla struttura di un modello. Anche nel caso di informazioni complete può esserci incertezza dovuta alle semplificazioni ed approssimazioni introdotte per rendere più trattabile l'analisi della informazione usata o il livello computazionale della stessa. Nel caso estremo si può essere incerti circa il grado di incertezza. Esiste quindi una varietà di tipi e di sorgenti di incertezza che si possono presentare in differenti situazioni e problemi della vita reale. Come misura dell'incertezza viene usata la nozione di probabilità, la cui teoria risulta essere il formalismo più frequentemente utilizzato per la sua quantificazione.

Numerosi contesti della vita reale in condizioni di incertezza, si configurano in termini di sistemi complessi e multidimensionali. La funzione prevalente dell'attività di ricerca applicata in queste aree è quella di cercare di comprendere le variabili che definiscono il sistema e le interrelazioni che fra queste si attivano. L'approccio fornito dalla Knowledge Discovery offre uno dei più avanzati contributi in questo ambito. In particolare il Knowledge Discovery in Database (KDD) indica l'intero processo di ricerca di nuova conoscenza formale procedendo da un database del fenomeno sotto studio. In letteratura esistono molti formalismi di rappresentazione della conoscenza estratta: in questo lavoro l'attenzione è rivolta alle reti Bayesiane, modelli grafici di dipendenza che usano la probabilità come componente quantitativa. Le reti Bayesiane sono il risultato della convergenza della Metodologia Statistica Bayesiana, che permette il passaggio dell'informazione contenuta nelle osservazioni sperimentali alla legge di probabilità che descrive il fenomeno, e le tecnologie dell'Intelligenza Artificiale, il cui intento è principalmente quello di permettere di trattare l'informazione mediante calcolatori. Le reti Bayesiane costituiscono oggi uno degli strumenti più completi e più coerenti per l'acquisizione, la rappresentazione e l'utilizzazione della conoscenza in condizioni di incertezza.

Una rete Bayesiana specifica una distribuzione di probabilità multivariata su un insieme di variabili aleatorie (modello probabilistico) attraverso due componenti:

- Un grafo diretto aciclico (DAG), detto struttura, in cui i nodi rappresentano le variabili aleatorie del dominio e gli archi, determinati da frecce dirette fra nodi, rappresentano le dipendenze condizionate fra le variabili che connettono;
- Un insieme di distribuzioni locali di probabilità, ciascuna associata ad una variabile aleatoria e condizionata dalle variabili corrispondenti ai nodi sorgenti degli archi entranti nel nodo che rappresenta la variabile, detti parenti.

Nella tesi si considerano solo variabili discrete e le distribuzioni locali assumono quindi la forma di tabelle di probabilità condizionata (CPT).

La mancanza di un arco fra due nodi riflette la loro indipendenza condizionata. L'ipotesi base per una rete Bayesiana afferma che ogni variabile è condizionatamente indipendente dai suoi non discendenti, dove un discendente di un nodo è definito o come un figlio del nodo oppure un discendente di uno dei suoi figli, dato i suoi parenti. Questa condizione, detta proprietà locale di Markov, porta alla specificazione di un'unica distribuzione di probabilità congiunta fattorizzabile in accordo con il grafo, permettendo una rappresentazione più compatta ed efficiente. Le relazioni di indipendenza evidenziate dalla proprietà di Markov implicano molte altre relazioni di indipendenza tra le variabili nella rete. La completa relazione tra indipendenza probabilistica e struttura grafica della rete è data dal concetto di d-separazione.

I vantaggi dell'uso delle reti Bayesiane come strumento di analisi per sistemi complessi sono molteplici. Le reti Bayesiane permettono di apprendere le relazioni che sussistono fra le variabili in modo da rappresentare la modularità nei sistemi complessi, ottenendo una rappresentazione grafica e strutturata intuitiva delle relazioni. Inoltre permettono di analizzare insiemi di dati, anche incompleti, poichè evidenziano la natura delle dipendenze e suggeriscono un modo naturale per codificarle. Utilizzando le tecniche di Statistica Bayesiana, le reti facilitano la combinazione del dominio di conoscenza ai dati, permettendo la possibilità di specificare dei giudizi soggettivi di esperti sul modello. Uno degli obiettivi principali delle reti Bayesiane è il calcolo della probabilità di un evento, inteso come insieme di assegnazioni di stato ad una o ad un insieme di variabili, che coinvolgono le variabili del dominio, condizionatamente ad ogni altro evento. Questo processo è chiamato *inferenza probabilistica*. In generale, quando nella rete vengono evidenziate molte dipendenze fra le variabili, l'inferenza probabilistica è NP-hard, poichè è necessario marginalizzare su un numero esponenziale di assegnazioni delle variabili. In letteratura sono stati proposti algoritmi per il calcolo dell'inferenza che includono sia metodi esatti che approssimati.

Nel contesto delle reti Bayesiane, la selezione del modello si traduce in termini di *apprendimento*, ovvero il processo di specificazione del grafo DAG, la struttura della rete, e della determinazione delle probabilità condizionate associate alle variabili del dominio, i parametri

della rete. Questi due tipi di apprendimento sono chiaramente non indipendenti, poichè l'insieme dei parametri necessari dipende dalla struttura assunta e viceversa. In letteratura sono sviluppati algoritmi di apprendimento automatico dai dati che si basano su differenti metodologie e che possono essere distinti in due sottogruppi principali, definiti come metodi Search & Score e Constraint-based, ognuno dei quali poggia su specifiche ipotesi e caratterizzazioni.

L'obiettivo di questa tesi è lo sviluppo di un nuovo metodo di apprendimento automatico delle reti Bayesiane dai dati, che si basa su una metodologia in grado di determinare strutture a cui corrisponde un processo inferenziale efficiente. La messa in opera di questo metodo porta alla costruzione e all'utilizzo di una rete Bayesiana per l'analisi di sistemi complessi che conserva la caratteristica di adattamento ai dati, presente negli altri metodi e che introduce la possibilità di poter fare inferenza probabilistica in modo computazionalmente semplificato.

La tesi è strutturata nel modo seguente.

Nel Capitolo 1 si definiscono formalmente le reti Bayesiane e si introduce la teoria su cui si basano gli sviluppi metodologici associati. Attraverso la presentazione di proprietà, teoremi e caratterizzazioni si definiscono le assunzioni fondamentali che permettono di rappresentare in modo efficiente la distribuzione di probabilità congiunta su un insieme di variabili, rendendo graficamente intuitive ed esplicite le dipendenze e le indipendenze che sussistono.

Nel Capitolo 2 si definisce il processo di inferenza probabilitica, ovvero il calcolo delle probabilità di un evento che coinvolge variabili del dominio condizionatamente a qualsiasi altro evento. A seguito di una panoramica degli approcci esistenti in letteratura relativamente al tipo di struttura assunta dalla rete, l'attenzione è rivolta alla metodologia *Junction Tree Propagation*, secondo la quale la propagazione dell'evidenza, definita come nuova informazione sullo stato delle variabili del dominio, può essere fatta in modo efficiente rappresentando la distribuzione di probabilità congiunta attraverso l'uso di un grafo indiretto detto *Junction Tree*.

Nel Capitolo 3 vengono presentate le tecniche esistenti in letteratura per l'apprendimento automatico delle reti Bayesiane. Per come è definita una rete Bayesiana, l'apprendimento si sviluppa attraverso metodi diversi, a seconda che il grafo della rete sia noto o non noto. Si presenta infine l'approccio innovativo proposto e sviluppato nella tesi che basa l'apprendimento della struttura di una rete Bayesiana su una misura collegata alla complessità del processo inferenziale. La misura definita nel nuovo approccio risulta essere un compromesso tra la bontà di adattamento del modello ai dati e la complessità del JT su cui verrà basata l'inferenza.

Nel Capitolo 4 vengono presentati i risultati sperimentali. Si è valutato il comportamento della nuova metodologia attraverso il confronto con i risultati ottenuti utilizzando alcuni fra i metodi più frequentemente utilizzati in letteratura: gli algoritmi di tipo greedy Hill-Climbing

basati su funzioni score BIC e BDe, e l'algoritmo K2. Si propongono inoltre le conclusioni ottenute applicando il metodo JT-Based a database reali, oggetto di analisi esplorativa in studi applicativi condotti in precedenza.

Nel Capitolo 5 si riassumono le conclusioni ottenute dallo studio e dall'analisi presentata nella tesi. Si propongono inoltre possibili temi e sviluppi, individuati all'interno del percorso di ricerca effettuato, da ampliare e approfondire in ricerche future.

Introduction

The term uncertainty is used in order to identify a variety of concepts. The uncertainty can arise from incomplete information, from not available information or from a disagreement of sources of information. The uncertainty can be referred to a variability of a phenomenon, to an amount or to a structure of a model. Even when the informations are complete, the uncertainty can arise from the reduction or the approximation which makes more tractable the analysis of the used information or its computational level. At least, the level of uncertainty can be uncertain. There exists, therefore, a variety of type and source of uncertainty, which can occur in different situations and problems of real life. As a measure of the uncertainty, the notion of probability is used, whose theory is the more frequently used formalism in order to quantify the uncertainty. Most of the real life domain in condition of uncertainty are complex and multidimensional systems. The prevalent function of applied research in these contexts is to try to understand the variables which define the system and their interrelationships. The approach provided from the Knowledge Discovery offers one of the most advanced contribution in this context. In particular, the Knowledge Discovery in Database (KDD) indicates the whole process of identification of knowledge from a database of the phenomenon in question. Several different representation formalisms are used to describe the extracted knowledge: in this thesis, we concentrate on Bayesian Networks, graphical models of dependance which use the probability as quantitative component. The Bayesian Networks are the result of the convergence of the Bayesian Statistics, which allow the flow of information from the experimental observations to the probability law which describes the phenomenon, and the technologies of Artificial Life, which allow to treat the information with the calculators. The Bayesian Networks are nowadays one of the most complete and coherent tool for the acquisition, the representation and the utilization of knowledge in condition of uncertainty.

A Bayesian Network specifies a multivariate probability distribution over a set of random variables (probabilistic model) through two components:

- A directed acyclic graph (DAG), the structure, in which the nodes represent the random variables and the arcs, directed arrows between nodes, represent conditional independences between the connected variables;

- A set of local probability distributions, each of them is associated to a random variable and conditionated to the set of its parents in the graph.

In the thesis only discrete variables are considered and the local distributions are in the form of conditional probability tables (CPT). The lack of an arc between two nodes reflects their conditional independence. Each variable in a Bayesian Network is conditionally independent from the set of all its non-descendents given the set of all its parents. This condition, named Markov Property, leads to a specification of a joint probability distribution, which is factorized according to the graph, allowing a more compact and efficient representation. The independence relationships highlighted by the Markov Property entail many other independence relationships between the variables in the network. The whole relation between probabilistic independence and graphical structure is given by the d-separation.

Using the Bayesian Networks as a tool of analysis in complex systems has many advantages. The Bayesian Networks allow to learn the relationships between the variables in order to represent the modularity in the complex systems, by giving an intuitive graphical representation of the relationships. Furthermore, they allow to analyse dataset, also incomplete, as they show the nature of the dependencies and they suggest a natural way for coding them. By using the Bayesian Statistics, the Bayesian Networks make easier the combination of the knowledge to the data, allowing to specify subjective judgments of experts. One of the main objective of the Bayesian Networks is to calculate the probability of an event, a set of assignments of the state to a variable or a set of variables, conditionally to every other event. This process is named *probabilistic inference*. Generally, when many dependencies between the variables are specified in the network, the probabilistic inference is NP-hard, since it is necessary to marginalized out an exponential number of variable assegnations. In literature, many algorithms are proposed which are exact or approximated methods.

In the Bayesian Networks context, model selection is the learning process of the DAG, the network's structure, and of the conditional probabilities associated to the variables in the domain. These two kind of learning are not independent, because the set of the parameters depends to the assumed structure and vice versa. In literature, many algorithms for the automatic learning from data are developed, which are based on different methodologies and which can be distinguished in two main subgroups: the Search & Score and the Constraint-based methods.

The objective of this thesis is to develop a new method of automatic learning from data

for the Bayesian Networks, based on a methodology which can identify the structure with an efficient associated process of probabilistic inference. To carry out this method is necessary to construct and to use a Bayesian Network which preservs the fitness to data, present in the other methods, and which introduces the possibility to do probabilistic inference in a easier computational manner.

The thesis is structured as follow.

In Chapter 1, the Bayesian Networks are formally defined and the theory on which are based the associated methodological developments are introduced. By the presentation of property, theorems and characterizations, the basic assumptions are defined which allow to represent in a efficient way the joint probability distribution of a set of variables, making the dependencies and the independencies of the system graphically intuitive and esplicit.

In Chapter 2, the process of probabilistic inference is defined, which is the calculation of the probability of an event conditionally to any other event. After a review of the existing approaches in literature based on the kind of considered structure, attention is given to *Junction Tree Propagation*, in which the propagation of evidence, defined as new information on the state of variables, is done in an efficient way by representing the joint probability distribution in an undirect graph named *Junction Tree*.

In Chapter 3, are presented the existing techniques in literature for the automatic learning of Bayesian Networks, the process by which the model selection is accomplished. The learning process develops in different ways depending on the graph is known or unknown. Finally, the proposed innovative approach is described which learns the structure of a Bayesian Network by a measure linked to the complexity of the inference process. The defined measure in the new approach is a compromise of the goodness of fit and the complexity of the JT on which the inference is carried out.

In Chapter 4, the experimental results are presented. The performance of the new method is evaluated through the comparison of the results obtained with some of the most frequently used methods in literature: the greedy Hill-Climbing algorithm with score function BIC, BDe and the K2 algorithm. The results of the application of the new method to real databases, used in previous explorative analysis, are presented.

In Chapter 5, the conclusions obtained in the thesis are recapitulated. Other possible themes and developments are proposed, in order to extend and to study in depth the obtained results.

Capitolo 1

Le reti Bayesiane

Nella tesi sono utilizzate le seguenti notazioni sintattiche.

Una variabile è denotata da una lettera maiuscola (per esempio, X, X_i, Θ) e il suo stato, ovvero il valore che essa assume, attraverso la corrispondente lettera minuscola (per esempio, x, x_i, θ). Un insieme di variabili è denotato da una lettera maiuscola in grassetto (per esempio, $\mathbf{X}$, $\mathbf{Y}$), e la corrispondente lettera minuscola in grassetto (per esempio, $\mathbf{x}$, $\mathbf{y}$) denota una assegnazione, o stato, per ogni variabile nel corrispondente insieme.

I termini, utilizzati nella tesi, che si riferiscono alla Teoria dei grafi, sono definiti formalmente in Appendice A.

Una rete Bayesiana è la rappresentazione grafica di un modello probabilistico, ovvero di una distribuzione di probabilità su un insieme di variabili $\mathbf{X} = \{X_1,X_2,\ldots,X_n\}$, e permette di rappresentare e analizzare un fenomeno oggetto di studio in condizioni di incertezza.

Le reti Bayesiane, denotate con $B = (G,\boldsymbol{\theta})$, sono definite attraverso la specificazione di due componenti:

(a) La componente **qualitativa**, un grafo diretto aciclico (DAG) $G = (\boldsymbol{\mathcal{V}},\boldsymbol{\mathcal{A}})$, detto *struttura* della rete, in cui i nodi $\boldsymbol{\mathcal{V}}$ sono in corrispondenza biunivoca con l'insieme $\mathbf{X} = \{X_1,X_2,\ldots,X_n\}$ di variabili aleatorie[1] e gli archi diretti $\boldsymbol{\mathcal{A}}$ sono coppie ordinate di elementi di $\boldsymbol{\mathcal{V}}$. Ogni arco, denotato con $X_i \rightarrow X_j$, rappresenta la dipendenza condizionata fra i nodi che connette; i parenti di un nodo X_i in G sono denotati da $Pa(X_i)$, i figli con $Ch(X_i)$[2].

(b) La componente **quantitativa**, un insieme di distribuzioni di probabilità condizionate

[1] In questa tesi si userà il termine variabile o nodo in modo intercambiabile.

[2] L'insieme dei parenti $Pa(X_i)$ e dei figli $Ch(X_i)$ di una variabile X_i, vengono indicati con le notazioni che specificano una variabile invece dell'usuale notazione in grassetto usata per gli insiemi di variabili.

(CPD) su $\mathbf{X}$, $\boldsymbol{\theta}$, dette *parametri* della rete. Le CPD locali $P(X_i|Pa(X_i))$, associate ad ogni variabile aleatoria e condizionate da ogni possibile combinazione dei valori assunti dall'insieme di parenti della variabile, sono specificate attraverso un insieme di parametri θ_i.

Le variabili aleatorie del dominio considerato possono essere discrete e continue. La tesi si focalizza sulle variabili discrete, caso più studiato ed analizzato in letteratura. In questo caso i parametri θ_i specificano distribuzioni di probabilità discrete, indicate con CPT. Per ogni CPT, si usa la notazione θ_{ijk} per indicare $P(X_i = x_k|Pa(X_i) = pa_j)$ ovvero la probabilità che la i-esima variabile si trovi nello stato k, dato che l'insieme dei suoi parenti è complessivamente nello stato j (indicato con pa_j).

I possibili valori delle variabili sono assunti mutuamente esclusivi ed esaustivi, il che significa che ogni variabile può assumere esattamente uno dei suoi possibili valori. Tipiche variabili discrete sono, ad esempio:

- le variabili booleane, che rappresentano proposizioni e che possono assumere solo valori binari di tipo {*Vero*, *Falso*} oppure {*Si*, *No*};
- le variabili ordinali, esempi delle quali sono giudizi di valore {*ottimo, buono, mediocre, cattivo, pessimo*}, o variabili originariamente misurabili su scala ad intervalli ma delle quali non si hanno tutte le informazioni {*giovane, maturo, vecchio*};
- le variabili a valori interi, come per esempio l'informazione legata al numero di componenti di una famiglia, che assume valori pari a 1, 2, . . .;
- le variabili continue opportunamente discretizzate, come ad esempio l'informazione legata all'altezza di un individuo, dicretizzata in classi significative per l'analisi[3].

In generale non c'è un limite al numero di valori discreti che può assumere una variabile.

La struttura della rete Bayesiana codifica le relazioni qualitative che sussistono tra le variabili. In particolare due nodi sono connessi attraverso un arco (freccia, legame diretto) se esiste una influenza tra i possibili valori che le variabili possono assumere. La forza delle relazioni che esistono tra le variabili è quantificata dalle distribuzioni di probabilità condizionata associate ad ogni nodo.

In Figura 1.1 è rappresentata la struttura di una rete Bayesiana.

[3]Esistono molti metodi per la discretizzazione delle variabili; la maggior parte degli strumenti di analisi basati sulle reti Bayesiane supportano la discretizzazione basata su uguale distanza o uguale frequenza degli intervalli determinati.

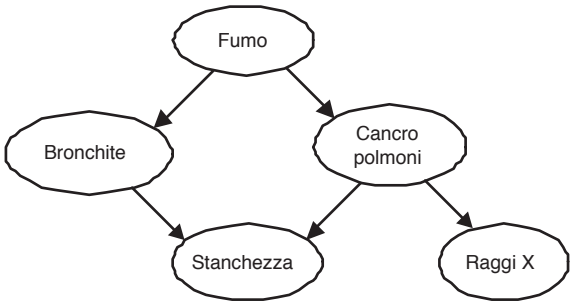


Figura 1.1. Esempio di rete Bayesiana

In questo esempio, i nodi corrispondono a variabili casuali booleane per ognuno dei quali viene specificata una tabella di probabilità condizionata (CPT) che codifica i parametri della rete. In Tabella 1.1 si evidenziano i parametri associati ad uno dei nodi del grafo associato alla rete Bayesiana dell'esempio. I nodi radice hanno associata una tabella di probabilità

C	B	S=0	S=1	
0	0	0.1	0.9	$\theta_{111} = P(S = 0\|C = 0,B = 0)$
0	1	0.7	0.3	$\theta_{121} = P(S = 0\|C = 0,B = 1)$
1	0	0.8	0.2	$\theta_{131} = P(S = 0\|C = 1,B = 0)$
1	1	0.9	0.1	$\theta_{141} = P(S = 0\|C = 1,B = 0)$

Tabella 1.1. Esempio di parametri associati ad una rete Bayesiana

marginale che rappresenta la probabilità a priori della variabile.

Se un nodo ha molti parenti o se i parenti di un nodo hanno numerosi stati possibili, la tabella di probabilità condizionata associata può essere molto grande. La dimensione di una CPT è infatti esponenziale nel numero dei parenti. Perciò, nel caso di una rete Bayesiana con variabili booleane, un nodo con p parenti richiede una CPT con 2^{p+1} probabilità (parametri della rete).

1.1 Indipendenze condizionate

Il tipo di connessione che unisce i nodi in una rete Bayesiana determina le indipendenze condizionate che sussistono tra le variabili della rete e il passaggio di informazione che si

attiva fra di esse, inteso come l'influenza che la conoscenza sullo stato di una variabile può avere su un'altra variabile.

Definizione 1.1 ***Indipendenza condizionata***
Siano X, Y e Z variabili casuali (o sottoinsiemi di variabili casuali $\mathbf{X}$,$\mathbf{Y}$ *e* $\mathbf{Z}$*). X e Y sono condizionatamente indipendenti dato Z se $P(X = x|Y = y,Z = z) = P(X = x|Z = z)$ per ogni valore x, y e z che le variabili possono assumere.*

L'indipendenza condizionata viene indicata con l'espressione $X \perp\!\!\!\perp Y|Z$. Si noti che la indipendenza non condizionata può essere trattata come un caso particolare della indipendenza condizionata considerando X, Y e $\emptyset$, in modo tale che $X \perp\!\!\!\perp Y \mid \emptyset$.

Si descrive l'insieme delle possibili situazioni che si possono verificare considerando tre variabili:

1. Connessione Seriale $X \rightarrow Y \rightarrow Z$: la conoscenza sullo stato della variabile X influenza la conoscenza su Y che a sua volta influenza Z. Inoltre, una informazione sullo stato di Z può influenzare la conoscenza su X attraverso Y; se lo stato di Y è noto, allora il passaggio dell'informazione è bloccato e X e Z diventano indipendenti condizionatamente a Y.

2. Connessione Convergente $X \rightarrow Y \leftarrow Z$: se non si hanno informazioni sullo stato di Y, eccetto quello che può essere dedotto dalla conoscenza sugli stati dei suoi parenti, X e Z risulteranno essere indipendenti e nessuna conoscenza sullo stato di uno di essi influenzerà la conoscenza sullo stato dell'altro. Se invece si ha informazione sullo stato di Y o di qualcuno tra i suoi figli, X e Z diventano dipendenti.

3. Connessione divergente $X \leftarrow Y \rightarrow Z$: se si conosce lo stato assunto da Y, non si ha passaggio di informazione fra i suoi figli X e Z, che risultano essere indipendenti.

In Figura 1.2 sono illustrate le possibili connessioni che si individuano fra gruppi di tre variabili nel grafo della rete Bayesiana rappresentata in Figura 1.1.

L'insieme delle indipendenze condizionate che sussistono in questa rete Bayesiana è riportato in Tabella 1.2.

In generale, il significato di una rete Bayesiana consiste in un insieme di affermazioni di indipendenze condizionate che sono implicate dalla struttura. Un approccio che permette di identificare le indipendenze è quello di "leggere" queste indipendenze dalla struttura della rete usando le regole della d-separazione descritta nella Sezione seguente.

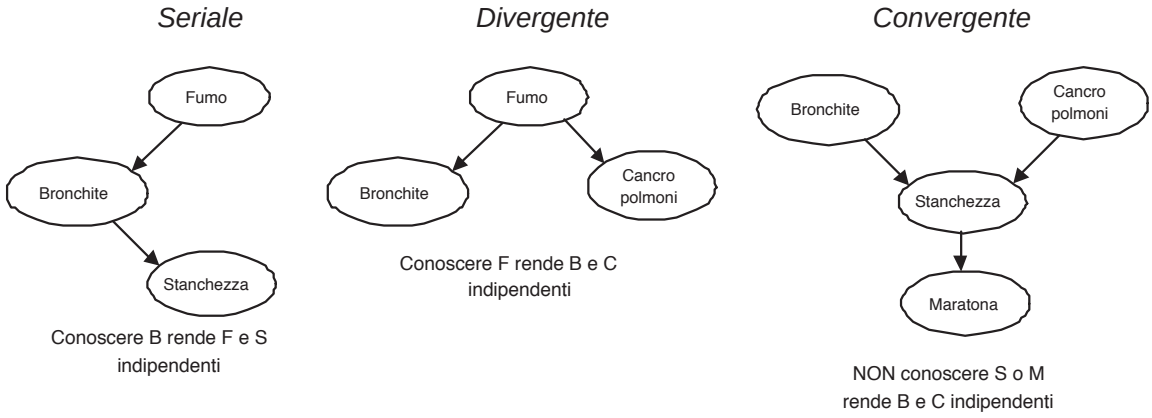


Figura 1.2. Esempio delle connessioni possibili tra tre variabili

Nodo	Parenti	Indipendenze condizionate
X	{C}	$X \perp\!\!\!\perp \{F,B,S\}\|C$
B	{F}	$B \perp\!\!\!\perp \{C,X\}\|F$
S	{B,C}	$S \perp\!\!\!\perp \{F,X\}\|\{B,C\}$
C	{F}	$C \perp\!\!\!\perp B\|F$

Tabella 1.2. Indipendenze condizionate

1.2 d-separazione

Pearl (1988) introduce la d-separazione come un criterio grafico che permette di identificare le indipendenze che sussistono tra le variabili, data la struttura G della rete Bayesiana. Una regola equivalente, alternativa, per l'identificazione delle indipendenze è proposta da Lauritzen et al. (1990).

Geiger et al. (1990) dimostrano che si è in grado di dedurre tutte le indipendenze che sono implicate logicamente dalla struttura della rete nel caso di dati discreti.

Meek (1997) prova che la d-separazione ha la proprietà di *completezza forte*, ovvero si possono dedurre tutte le combinazioni degli insiemi disgiunti e/o congiunti di affermazioni di indipendenza, implicate logicamente dalla struttura.

Definizione 1.2 ***d-separazione***
I nodi X e Y sono d-separati se in ogni cammino tra X e Y esiste un nodo Z tale che:

- Z è in connessione seriale o divergente e Z è noto,
oppure
- Z è in connessione convergente e né Z né nessuno dei suoi discendenti è noto.

Se X e Y non sono d-separati, si dicono *d-connessi*.

Il seguente teorema lega la nozione di d-separazione con l'indipendenza condizionata :

Teorema 1.1 *(Verma e Pearl, 1988)*
Se i nodi X e Y sono d-separati da Z allora X e Y sono condizionatamente indipendenti dato Z, $X \perp\!\!\!\perp Y \mid Z$.

Questo risultato permette di limitare i calcoli legati alle probabilità delle variabili attraverso le proprietà codificate dal grafo.

Siano X e Y, ad esempio, d-separati da Z e sia noto lo stato assunto da Z. Se è noto il valore di $P(X|Z)$ e si ottiene una informazione sullo stato di Y, il teorema permette di specificare $P(X|Z)$ come valore di $P(X|Z,Y)$.

Quindi, i risultati descritti per la nozione di d-separazione e per il blocco dell'informazione, deducibili graficamente attraverso la struttura, risultano essere validi anche per la rappresentazione probabilistica, quantitativa, sottostante il modello.

1.3 Proprietà di Markov

La proprietà di Markov è fondamentale per una distribuzione di probabilità modellata attraverso una rete Bayesiana[4].

Definizione 1.3 ***Proprietà di Markov***
Una variabile è condizionatamente indipendente dei suoi non discendenti dati i suoi parenti

$$X_i \perp\!\!\!\perp Nd(X_i) \mid Pa(X_i) \textit{ per ogni } X_i \in \mathbf{X}$$

La proprietà di Markov descrive un insieme minimo di relazioni di indipendenza, che esiste tra ogni nodo e i suoi non discendenti, data la struttura della rete Bayesiana.

Attraverso questa proprietà e l'utilizzo di assiomi validi per le indipendenze condizionate, (Pearl, 1988), si specificano tutte le relazioni di indipendenza che caratterizzano il dominio oggetto di studio.

[4]Pearl (1988) denota come una *Mappa d'Indipendenza* (I-Map) per dominio di variabili $\mathbf{X}$ un DAG che soddisfa alla proprietà di Markov per una distribuzione $P(\mathbf{X})$.

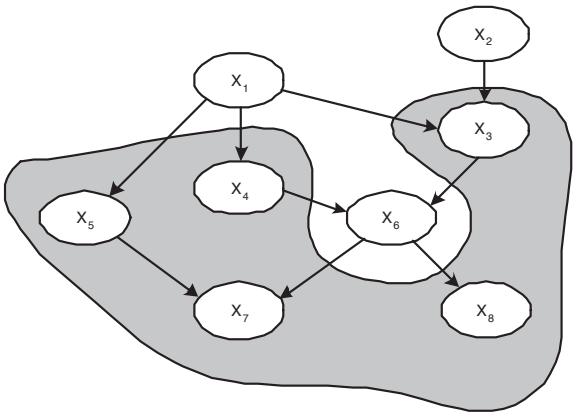


Figura 1.3. Markov Blanket della variabile X_6

Definizione 1.4 ***Markov Blanket***
Per ogni variabile X_i, il Markov Blanket $MB(X_i)$ è l'insieme delle variabili in $\mathbf{X}$ tale che X_i è condizionatamente indipendente da tutte le altre variabili del dominio dato il suo Markov Blanket.

In altri termini, $MB(X_i)$ blocca, o d-separa, la variabile X_i da ogni altra variabile che non appartiene a $MB(X_i) \cup \{X_i\}$, rendendola probabilisticamente indipendente da questi nodi. Nel contesto delle reti Bayesiane, il Markov Blanket di una variabile è identificabile dal grafo: esso consiste dell'insieme dei suoi parenti, dei suoi figli e dei parenti dei suoi figli. In Figura 1.3 è presentato un esempio di Markov Blanket per una variabile di riferimento.

Non tutte le dipendenze e le indipendenze condizionate valide nella struttura sono valide anche nella distribuzione di probabilità che si vuole modellare. Si ha la seguente:

Definizione 1.5 ***Faithfulness***
Un grafo G e una distribuzione di probabilità P sono Faithful[5] *se tutte e solo le relazioni di indipendenza valide in P sono quelle implicate dalla proprietà di Markov in G.*

Il seguente teorema stabilisce un criterio per identificare la Faithfulness:

Teorema 1.2 *(Geiger e Pearl, 1990)*
Un grafo G e una distribuzione di probabilità P sono Faithful se e solo se tutte e sole le indipendenze condizionate in P sono quelle identificate dalla d-separazione in G.

[5] *In Pearl (1988), G è una* Mappa Perfetta *di $P(\mathbf{X})$.*

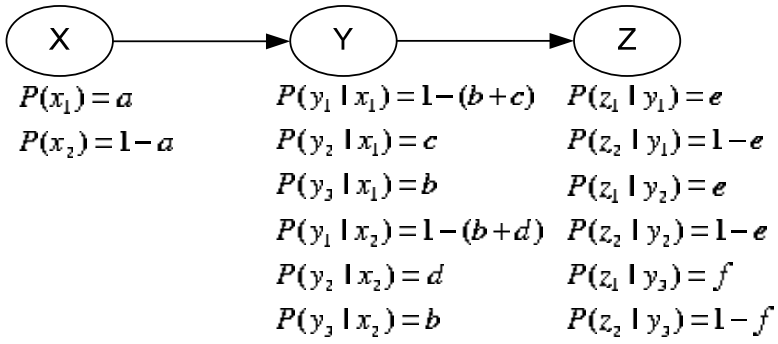


Figura 1.4. Esempio di grafo e distribuzione di probabilità *non* Faithful

Una distribuzione di probabilità P, che soddisfa la proprietà di Markov in G, può avere indipendenze condizionate che non sono identificate dalla d-separazione. In Figura 1.4 sono rappresentate una rete Bayesiana e le corrispondenti distribuzioni di probabilità locali associate ad ogni nodo; vale $X \perp\!\!\!\perp Y$, ma X non è d-separato da Y e quindi non viene soddisfatta la condizione di Faithful.

Meek (1997) dimostra che quasi tutte le distribuzioni discrete, in particolare le distribuzioni di probabilità multinomiali, associate ad una data struttura sono Faithful, ovvero le relazioni di indipendenza che valgono nella distribuzione sono tutte e solo quelle implicate dalla struttura della rete. In questo modo, una volta appresa la struttura della rete Bayesiana, si determinano attraverso la d-separazione le indipendenze condizionate e si assume che queste siano in grado di specificare perfettamente la distribuzione di probabilità congiunta sottostante i dati.

1.4 Rappresentazione della distribuzione di probabilità congiunta

Le reti Bayesiane sono considerate essere la rappresentazione di una distribuzione di probabilità congiunta. In particolare, la distribuzione di probabilità P associata alle variabili $\mathbf{X} = \{X_1, X_2, \ldots, X_n\}$, è fattorizzabile in distribuzioni di probabilità locali che coinvolgono un nodo e l'insieme dei suoi parenti.

Si consideri la distribuzione congiunta $P(\mathbf{X}) = P(X_1, X_2, \ldots, X_n)$.

La *Regola a catena* della Teoria della Probabilità permette di fattorizzare questa distribuzione nel seguente modo:

$$\begin{aligned} P(X_1,X_2,\ldots,X_n) &= P(X_n|X_1,\ldots,X_{n-1})P(X_{n-1}|X_1,\ldots,X_{n-2})\cdots P(X_2|X_1)P(X_1) \\ &= \prod_{i=1}^{n} P(X_i|X_1,\ldots,X_{i-1}) \end{aligned}$$

Per la proprietà di Markov, la struttura di una rete Bayesiana implica che il valore di un particolare nodo è condizionato solo dal valore che assumono i suoi parenti. Ciò porta alla specificazione di una fattorizzazione ricorsiva della distribuzione P, in accordo con il grafo G della rete, tale che:

$$P(X_1,X_2,\ldots,X_n) = \prod_{i=1}^{n} P(X_i|Pa(X_i))$$

In questo modo si ottiene una rappresentazione più compatta della distribuzione di probabilità congiunta delle variabili del dominio oggetto di studio, che semplifica la descrizione della probabilità di ogni insieme di informazioni sullo stato delle variabili.

Si consideri la rete Bayesiana, già rappresentata in termini di Markov Blanket per una variabile, nella Figura 1.3.

Supponenedo che tutte le variabili siano booleane, la distribuzione di probabilità congiunta sulle variabili è fattorizzabile nel seguente modo:

$$\begin{aligned} P(X_1,X_2,\cdots,X_8) &= P(X_1)\cdot P(X_2)\cdot P(X_3|X_1,X_2)\cdot P(X_4|X_1)\cdot P(X_5|X_1) \\ &\quad \cdot P(X_6|X_3,X_4)\cdot P(X_7|X_5,X_6)\cdot P(X_8|X_6) \end{aligned}$$

E' necessario dunque specificare un numero pari a 20 parametri, ricordando che un parametro rappresenta la probabilità che la variabile X_i sia nel suo stato k-esimo condizionatamente al fatto che l'insieme dei suoi parenti è complessivamente nello stato j-esimo, invece di un numero di parametri pari a $2^8 = 256$.

Quando si utilizza una rete Bayesiana per modellare una distribuzione di probabilità congiunta, non necessariamente una sola struttura è in grado di codificare la distribuzione di interesse.

Si introduce in questo contesto la nozione di classe di equivalenza delle reti Bayesiane (Verma e Pearl, 1991).

Si considerino i tre DAG in Figura 1.5. Nel DAG (a), i parametri da stimare sono $P(X)$, $P(Y|X)$ e $P(Y|Z)$. Per il DAG (b) i parametri sono $P(Y)$, $P(X|Y)$ e $P(Z|Y)$. Dalla definizione di probabilità condizionata si ha che $P(X)P(Y|X) = P(X,Y) = P(Y)P(X|Y)$, il che implica che la distribuzione di probabilità congiunta codificata dai due DAG è equivalente. Ciò significa che i parametri del primo modello determinano completamente i parametri

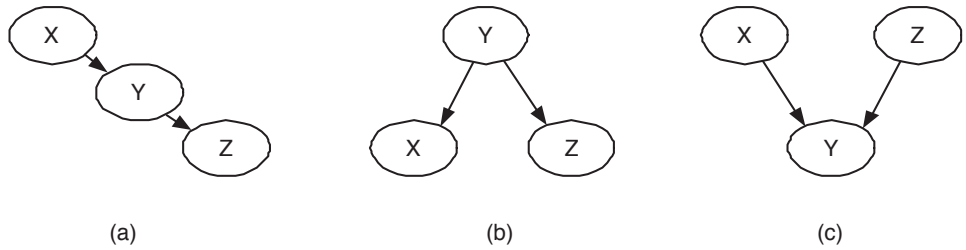


Figura 1.5. Esempio di DAG equivalenti (a) e (b), e non equivalenti (c)

del secondo e vicevera. I parametri da stimare del DAG (c) sono $P(X)$, $P(Z)$ e $P(Y|X,Z)$ ed è chiaro che non possono essere ottenuti dai parametri dei due modelli precedenti. Dunque DAG (c) è diverso da i DAG (a) e (b).

Definizione 1.6 ***Equivalenza in distribuzione ed in indipendenza***
Due DAG G e G' sono equivalenti in distribuzione *se per ogni rete Bayesiana $B = (G,\boldsymbol{\theta})$ esiste una rete $B' = (G', \boldsymbol{\theta}')$ tali che B e B' definiscono la stessa distribuzione di probabilità e viceversa. Due DAG G e G' sono* equivalenti in indipendenza *se le relazioni di indipendenza codificate dai due DAG sono uguali.*

Due DAG G e G' sono *equivalenti* se sono sia equivalenti in distribuzione che equivalenti in indipendenza. Poichè la relazione di equivalenza è riflessiva, simmetrica e transitiva, la relazione definisce un insieme di *classi di equivalenza* sulle strutture della rete.

Nel contesto delle reti Bayesiane, Verma e Pearl (1991) definiscono due caratteristiche in grado di determinare la classe di equivalenza di un DAG.

Definizione 1.7 ***Scheletro***
Lo scheletro di un DAG è il grafo indiretto che risulta ignorando la direzionalità di ogni suo arco.

Definizione 1.8 ***V-struttura***
Una V-struttura in un DAG G è una terna ordinata di nodi (X,Y,Z) tale che:

1. *G contiene gli archi $X \rightarrow Y$ e $Z \rightarrow Y$;*
2. *X e Z non sono adiacenti in G.*

Teorema 1.3 *(Verma e Pearl, 1991)*
Due DAG sono equivalenti se e solo se hanno lo stesso scheletro e la stessa V-struttura.

Questa caratterizzazione delle reti Bayesiane può essere tenuta in considerazione nella fase di apprendimento della rete; esistono infatti alcuni algoritmi, evidenziati nel Capitolo 3, che utilizzano questa proprietà per ridurre lo spazio di ricerca delle strutture della rete Bayesiana, nel caso di apprendimento automatico con dati completi.

Per maggiori approfondimenti riguardanti la teoria delle reti Bayesiane, si vedano, fra gli altri, Buntine (1996), Heckerman (1996), Jensen (1996), Lauritzen (1996) e Neapolitan (2004).

Capitolo 2

Inferenza nelle reti Bayesiane

Uno degli obiettivi della costruzione delle reti Bayesiane è, data l'osservazione corrente sullo stato di alcune tra le variabili del dominio, rispondere a quesiti sulla distribuzione di probabilità di alcuni valori di variabili di interesse; questo processo è detto *inferenza probabilistica* o *Belief Updating*.

Una rete Bayesiana, completamente specificata, contiene l'informazione necessaria per rispondere a tutti i quesiti probabilistici circa le variabili d'interesse. Il meccanismo che permette di trarre conclusioni in una rete Bayesiana è detto *propagazione dell'evidenza*, dove con evidenza si intende l'informazione corrente di cui si dispone. La propagazione consiste nell'aggiornare le distribuzioni di probabilità delle variabili in accordo alla nuova evidenza disponibile (Castillo et al., 1997, Cowell et al., 1999).

Siano $\mathbf{X} = \{X_1, \ldots, X_n\}$ un insieme di variabili aleatorie discrete e sia $P(\mathbf{X})$ una distribuzione di probabilità definita su $\mathbf{X}$.

Prima che sia disponibile qualsiasi evidenza, il processo di propagazione consiste nel calcolo delle distribuzioni di probabilità marginali $P(X_i = x_i)$, indicate semplicemente anche con $P(x_i)$, per ogni $X_i \in \mathbf{X}$.

Si supponga successivamente che una qualche evidenza sia resa disponibile, ossia è noto che un sottoinsieme di variabili $\mathbf{E} \in \mathbf{X}$ assume complessivamente lo stato $\mathbf{e}$ (si ha $X_i = e_i$ con $X_i \in \mathbf{X}$ e $e_i \in \mathbf{e}$). In questa situazione, la propagazione dell'evidenza permette di calcolare le probabilità condizionate $P(x_i|e_i)$.

La propagazione dell'evidenza può avvenire nei seguenti modi:

- Dall'alto verso il basso: l'evidenza si ha nei parenti e negli ascendenti e si vogliono calcolare le probabilità dei figli o dei discendenti;
- Dal basso verso l'alto: l'evidenza si ha nei figli e nei discendenti e si vogliono calcolare le nuove probabilità dei parenti o degli ascendenti.

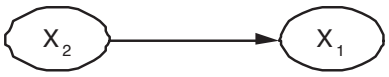


Figura 2.1. Esempio di rete Bayesiana

Esistono due tipi di evidenza:

- Evidenza *hard*: si conosce lo stato assunto da una o più variabili (si dice che la variabile è *instantiated*);
- Evidenza *soft*: si fanno affermazioni sullo stato assunto da una o più variabili (per esempio, si ha l'informazione che la variabile assume valori inferiori ad una certa soglia).

Le reti Bayesiane, rappresentando in modo compatto ed efficiente la distribuzione di probabilità congiunta sulle variabili del dominio, semplificano il processo di inferenza probabilistica, aggiornando le probabilità del modello attraverso la regola di Bayes.

Si consideri la rete Bayesiana in Figura 2.1. Per le variabili X_1 e X_2, l'applicazione della regola di Bayes comporta:

$$P(X_1|X_2) = \frac{P(X_2|X_1)P(X_1)}{\sum_i P(X_2|X_1 = x_i)P(X_1 = x_i)}$$

Successivamente si osserva, per esempio, che la variabile X_2 è nello stato x_j (si ha evidenza di tipo hard). Applicando l'equazione precedente si calcola la distribuzione di probabilità $P(X_1|X_2 = x_j)$ nel seguente modo:

$$P(X_1|X_2 = x_j) = \frac{P(X_2 = x_j|X_1)P(X_1)}{\sum_i P(X_2 = x_j|X_1 = x_i)P(X_1 = x_i)}$$

Calcoli simili possono essere fatti per reti più ampie permettendo l'analisi di differenti aspetti di un fenomeno oggetto di studio. L'aggiornamento delle probabilità che si basa sul metodo descritto precedentemente, è trattabile solo la rete è piccola e se ogni nodo rappresenta variabili che possono assumere solo pochi valori. Inoltre se si assume di avere evidenza su un insieme di variabili, è difficile assicurare un aggiornamento consistente delle probabilità di ogni nodo, in accordo con l'insieme di evidenza considerata.

Per risolvere questo problema sono state sviluppate numerose tecniche di propagazione dell'evidenza, che permettono il calcolo dell'inferenza in modo corretto ed efficiente.

A seconda del tipo di struttura assunta dalla rete, sono stati sviluppati in letteratura algoritmi che permettono di eseguire inferenza esatta o approssimata. Per alcune reti, l'inferenza

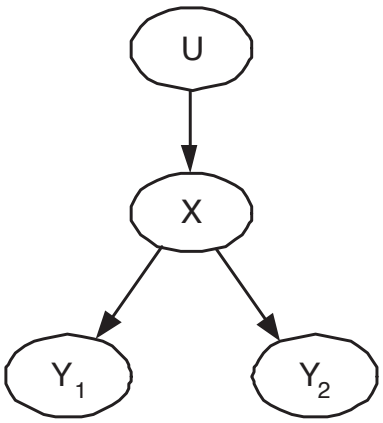


Figura 2.2. Esempio di struttura ad albero

esatta è computazionalmente intrattabile e si devono usare algoritmi di inferenza approssimata. In generale, sia l'inferenza esatta che quella approssimata è computazionalmente complessa: si prova infatti che entrambe sono, nel peggiore dei casi, NP-hard (Cooper, 1990; Dagum e Horvitz, 1993). In pratica, la velocità dell'inferenza dipende da fattori quali la struttura della rete, la numerosità dei cicli indiretti presenti e dal posizionamento dell'evidenza considerata. Nella Sezione seguente si presenta una panoramica dei metodi presenti in letteratura. Ci si focalizza, in particolare, sugli algoritmi di inferenza esatta.

2.1 Approcci esistenti in letteratura

2.1.1 Inferenza negli Alberi

Le indipendenze condizionate implicate dalla proprietà di Markov possono essere utilizzate per fare inferenza in una rete Bayesiana.

Utilizzando le indipendenze locali, Pearl (1982) sviluppa un algoritmo chiamato *Message Passing* per reti con struttura ad albero, ovvero un DAG in cui esiste un unico nodo, detto radice, che non ha parenti mentre ogni altro nodo ha esattamente un parente ed è un discendente del nodo radice (Figura 2.2).

Dato un insieme $\mathbf{e}$ di valori assunti dall'insieme $\mathbf{E} \subset \mathbf{X}$ di variabili dette *instantiated*, l'algoritmo determina $P(X_i|\mathbf{e})$ per tutti i valori assunti da ogni X_i della rete in esame. Ciò avviene inizializzando i messaggi che ogni variabile *instantiated* manda ai suoi nodi vicini,

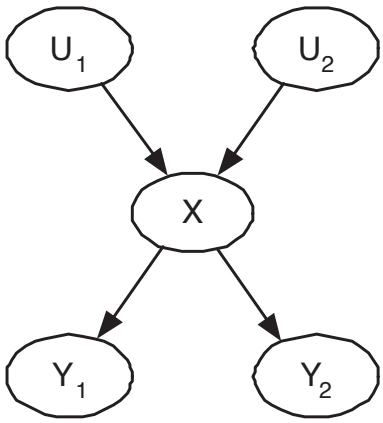


Figura 2.3. Esempio di struttura a polialbero

che a loro volta mandano ai propri vicini. L'aggiornamento non dipende dall'ordine con cui si inizializzano questi messaggi.

La definizione dei messaggi che si trasmettono le variabili e la procedura che permette la propagazione dell'evidenza viene descritta nella Sezione seguente: l'algoritmo proposto da Pearl (1982) è generalizzato in Pearl (1988), in cui, la struttura di riferimento è un polialbero. Un albero è infatti un caso particolare di un polialbero, in cui ogni nodo ha esattamente un parente.

2.1.2 Inferenza nei Polialberi

Un DAG è un polialbero, definito anche rete a connessione singola, se esiste al massimo una catena tra ogni coppia di nodi (Figura 2.3). La differenza tra un albero e un polialbero è che, in quest'ultimo, un nodo può avere più di un parente.

Pearl (1988) propone l'algoritmo *Message Passing* generalizzato: quest'ultimo è un algoritmo iterativo che permette il passaggio dell'informazione sia dall'alto verso il basso che dal basso verso l'alto attraverso la definizione di messaggi da un nodo aggiornato verso i suoi parenti e i suoi figli. In ogni iterazione i messaggi si diffondono attraverso l'intera rete fino a che l'aggiornamento delle probabilità non è completato.

Poichè la struttura considerata è un polialbero, dalla proprietà di Markov si ha che l'aggiornamento della probabilità di un nodo può essere calcolato localmente: i messaggi dai parenti e dai figli di un nodo rappresentano tutta l'evidenza relativa al sottografo considerato.

Sia $\mathbf{E}$ l'evidenza disponibile. $\mathbf{E}$ può essere scomposta in due sottoinsiemi:

- E_i^+, il sottoinsieme di $\mathbf{E}$ che può essere raggiunto da X_i attraverso i suoi parenti;
- E_i^-, il sottoinsieme di $\mathbf{E}$ che può raggiunto da X_i attraverso i suoi figli.

Da questa scomposizione segue che:

$$P(x_i|\mathbf{e}) = P(x_i|e_i^-,e_i^+) = \frac{1}{P(e_i^-,e_i^+)} P(e_i^-,e_i^+|x_i)P(x_i)$$

Poichè X_i separa E_i^- da E_i^+ nel polialbero, vale l'affermazione di indipendenza condizionata $E_i^- \perp\!\!\!\perp E_i^+ \mid X_i$; da ciò segue:

$$\begin{aligned} P(x_i|\mathbf{e}) &= \tfrac{1}{P(e_i^-,e_i^+)} P(e_i^-|x_i)P(e_i^+|x_i)P(x_i) \\ &= \tfrac{1}{P(e_i^-,e_i^+)} P(e_i^-|x_i)P(x_i,e_i^+) \\ &= k\lambda_i(x_i)\pi_i(x_i) \end{aligned}$$

.

Per calcolare le funzioni $\lambda_i(x_i)$ e $\pi_i(x_i)$, si suppone che un nodo X_i abbia p parenti, $\mathbf{U} = \{U_1, \ldots, U_p\}$, e c figli, $\mathbf{Y} = \{Y_1, \ldots Y_c\}$. L'evidenza E_i^+ può essere partizionata in p componenti disgiunte, una per ogni parente di X_i:

$$E_i^+ = \{E_{U_1X_i}^+, \ldots, E_{U_pX_i}^+\}$$

dove l'evidenza $E_{U_jX_i}^+$ è il sottoinsieme di E_i^+ relativo all'arco $U_j \rightarrow X_i$.

In modo simile, l'evidenza E_i^- può essere partizionata in c componenti disgiunte.

Sia $\mathbf{u} = \{u_1,...,u_p\}$ l'insieme degli stati osservati per i parenti $\mathbf{U}$ di X_i. Si ha:

$$\begin{aligned} \pi_i(x_i) &= P(x_i,e_i^+) \\ &= \textstyle\sum_u P(x_i,u \cup e_i^+) \\ &= \textstyle\sum_u P(x_i|u \cup e_i^+)P(u \cup e_i^+) \\ &= \textstyle\sum_u P(x_i|u \cup e_i^+)P(u \cup e_{U_1X_i}^+ \cup \ldots \cup e_{U_pX_i}^+). \end{aligned}$$

Poichè $\{U_j,E_{U_jX_i}^+\}$ è indipendente da $\{U_k,E_{U_kX_i}^+\}$ per ogni $j \neq k$, si ha:

$$\pi_i(x_i) = \sum_u P(x_i|u \cup e_i^+) \prod_{j=1}^{p} P(u_j \cup e_{U_jX_i}^+),$$

e

$$\pi_{U_jX_i}(u_j) = P(u_j \cup e_{U_jX_i}^+)$$

è il *messaggio* π che il nodo U_j manda al figlio X_i.

Analogamente si definisce il *messaggio* λ_i che il nodo X_i manda ad ogni figlio appartenete all'insieme $\mathbf{Y}$. La combinazione del passaggio dei messaggi definiti per tutti i parenti e i figli del nodo X_i, come mostrato in Figura 2.4, dà luogo all'algoritmo definito in Pearl (1988), al quale si rimanda per una descrizione dettagliata.

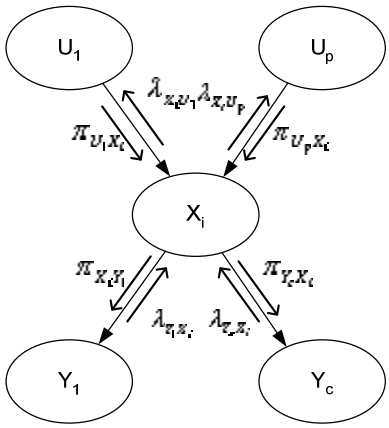


Figura 2.4. Passaggio di messaggi per un nodo

2.1.3 Inferenza nelle reti a connessioni multiple

L'algoritmo *Message Passing* è un algoritmo efficiente nel senso che i calcoli effettuati durante la propagazione del messaggio sono locali: l'aggiornamento della probabilità e i relativi nuovi messaggi uscenti sono calcolati usando messaggi entrati e le tabelle di probabilità condizionata associate ai nodi. Nonostante ciò, si ha che questo algoritmo è computazionalmente intrattabile quando ogni nodo presenta un ampio numero di parenti.

Inoltre molte reti Bayesiane, specialmente nel caso in cui si stia analizzando un sistema complesso, presentano strutture più articolate dei polialberi. In questi casi l'algoritmo *Message Passing* non può essere applicato.

Nel più generale dei casi, la struttura della rete è un DAG, il che significa che almeno due nodi sono connessi da più di un cammino indiretto. Gli algoritmi di inferenza basati sul *Clustering* generalizzano l'algoritmo *Message Passing* in modo da poter fare inferenza probabilistica considerando strutture di tipo DAG. L'idea base è quella di trasformare la rete Bayesiana in un polialbero probabilisticamente equivalente unendo i nodi e eliminando i cammini multipli tra nodi lungo i quali si può propagare l'evidenza.

Il metodo di inferenza esatta più utilizzato in questo contesto è l'algoritmo *Junction Tree Propagation*, sviluppato da Lauritzen e Spiegelhalter (1988) e perfezionato da Jensen et al. (1990). Questo metodo, descritto in dettaglio successivamente, trasforma una rete a connessioni multiple in un albero, i cui nodi sono gruppi di variabili della rete originale e applica tecniche di passaggio di messaggi tra nodi dell'albero.

L'approccio detto *Arc reversal/Node reduction* sviluppato da Shachter (1986) propone di applicare una sequenza di operatori alla rete che inverte i legami attraverso la regola di Bayes.

L'algoritmo *Variable Elimination* (Zhang e Poole, 1996) elimina le variabili di non interesse attraverso sommatorie, con una complessità basata sul numero di moltiplicazioni e somme che si eseguono durante l'intero processo.

Pearl (1988) presenta un algoritmo di inferenza per reti a connessioni multiple chiamato *Loop Cutset Conditioning*, che modifica le connessioni nella rete rendendola a connessioni singole in modo da poter applicare le tecniche di *Message Passing*.

2.2 Junction Tree Propagation

Uno dei metodi più efficienti per l'inferenza probabilistica proposti in letteratura è la *propagazione basata sui Junction Tree*. Questo metodo appartiene alla classe più ampia dei metodi per l'inferenza basati sul *Clustering*, per i quali l'idea generale è quella di trasformare la rete Bayesiana in un polialbero in grado di rappresentare la stessa distribuzione di probabilità congiunta e successivamente utilizzare questa nuova struttura per fare inferenza.

In particolare, questo metodo trasforma la rete Bayesiana in un grafo indiretto detto *Junction Tree* (JT), e definiti i *potenziali* associati ad ogni nodo del JT, utilizza le distribuzioni di probabilità ad essi corrispondenti per il processo d'inferenza.

Definizione 2.1 Struttura del Junction Tree
Data una rete Bayesiana su un dominio $\mathbf{X} = \{X_1, X_2, \ldots, X_n\}$*, la struttura del Junction Tree corrispondente è un albero indiretto i cui nodi sono clusters* C_i *di variabili*[1]*, detti* clique*, appartenenti ad* $\mathbf{X}$ *tali che:*

- *per ogni* $X_k \in \mathbf{X}$, $F_{X_k} = X_k \cup Pa(X_k)$ *appartiene ad almeno un cluster;*
- *dati due cluster* C_1 *e* C_2*, ogni nodo nel cammino che li unisce contiene la loro intersezione* $C_1 \cap C_2$*;*
- *ad ogni arco dell'albero è associato un* separatore S *che contiene le variabili dell'intersezione tra nodi vicini.*

Una clique è dunque un sottografo *completo*[2] e *massimale*[3] della rete originaria.

La componente quantitativa del JT, è specificata attraverso la nozione di *potenziale* associato ad ogni clique e ad ogni separatore.

[1] *Nelle notazioni legate alla teoria dei Junction Tree, i cluster* C_i *vengono indicati nel modo in cui si specifica una variabile invece della notazione in grasseto usata per gli insieme di variabili.*

[2] Un sottografo è completo se è presente un arco tra ogni coppia di nodi.

[3] Un sottografo completo è massimale se non è contenuto in nessun altro sottografo completo.

Definizione 2.2 Potenziale
Un potenziale è una funzione a valori reali definita su un insieme di variabili $C = \{X_1, \ldots, X_n\}$

$$\phi_C : C \longrightarrow \mathbb{R}$$

per la quale valgono le seguenti operazioni:

1. Marginalizzazione*:* $\phi_{C_2} = \phi_{C_1} \setminus C_2 = \sum_{C_1 \setminus C_2} \phi_{C_1}$ *se* $C_2 \subseteq C_1$
2. Moltiplicazione*:* $\phi_{C_3} = \phi_{C_1} \cdot \phi_{C_2}$ *se* $C_3 = C_1 \cup C_2$

Per esempio, consideriamo gli insiemi $C_3 = \{X_1,X_2,X_3\}$, $C_1 = \{X_1,X_2\}$ e $C_2 = \{X_2,X_3\}$. Per il processo di marginalizzazione si ha che $\phi_{C_1} = \sum_{X_3} \phi_{C_1}$, e per il processo di moltiplicazione $\phi_{C_3} = \phi_{X_1X_2X_3} = \phi_{X_1X_2}\phi_{X_2X_3} = \phi_{C_1}\phi_{C_2}$. Si noti che ϕ_{C_1} e ϕ_{C_2} sommano a 1, ma ciò non vale necessariamente per ϕ_{C_3}.

I potenziali di un JT non sono specificati in modo arbitrario; essi infatti devono soddisfare le seguenti condizioni:

- *Consistenza locale*: per ogni clique C e per ogni separatore S adiacente

$$\phi_S = \phi_C \setminus S = \sum_{C \setminus S} \phi_C.$$

- *Consistenza Globale*: la distribuzione di probabilità congiunta delle variabili del dominio è fattorizzabile in

$$P(\mathbf{X}) = \frac{\prod_i \phi_{C_i}}{\prod_i \phi_{S_i}}.$$

Se un JT soddisfa queste proprietà, segue che per ogni clique (o separatore) C si ha che $\phi_C = P(C)$. Usando questa proprietà, si può calcolare la distribuzione di ogni variabile X_k del dominio, usando ogni cluster (o separatore) C che contiene X_k nel seguente modo:

$$P(X_k) = \sum_{C \setminus \{X_k\}} \phi_C.$$

L'algoritmo di Junction Tree Propagation si sviluppa in tre passi principali:

1. **Trasformazione della struttura**
   - **a.** Moralizzazione
   - **b.** Triangolarizzazione
   - **c.** Costruzione del JT

2. **Inizializzazione**

   a. Fissare i potenziali

   b. Propagazione dei potenziali

3. **Aggiornamento delle probabilità**

   a. Inserimento dell'evidenza nel JT

   b. Propagazione dei potenziali

Il passo 1. è la componente grafica dell'algoritmo, mentre i passi 2. e 3. sono la componente numerica. Per la stessa rete Bayesiana possono esistere differenti JT, ma una volta determinata la struttura e inizializzato i potenziali associati (passi 1. e 2. dell'algoritmo) non è più necessario eseguire ancora queste operazioni per aggiornare le probabilità.

2.2.1 Moralizzazione e Triangolarizzazione

Sia G un DAG di una rete Bayesiana. Il **grafo morale** G^M corrispondente a G è un grafo indiretto costruito nel seguente modo (Lauritzen e Spiegelhalter, 1988):

- per ogni $X_k \in \mathbf{X}$ *sposare i parenti* ovvero per ogni Y,$Z \in Pa(X_k)$ aggiungere, dove non gia esistente, un arco $Y - Z$
- Rendere tutti gli archi indiretti

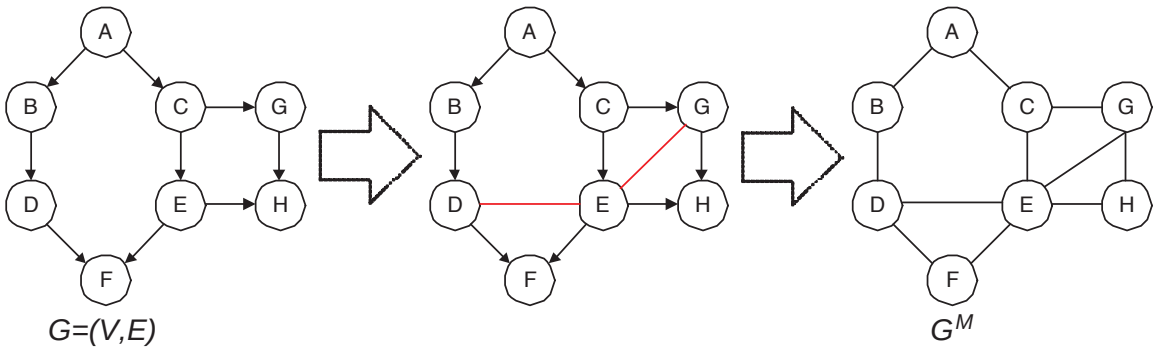


Figura 2.5. Costruzione del grafo morale

In Figura 2.5 si illustra il processo di moralizzazione. Gli archi indiretti aggiunti sono detti *archi morali* e sono mostrati in rosso nella figura.

Definizione 2.3 ***Grafo triangolato***
Un grafo indiretto è detto triangolato se ogni suo ciclo di lunghezza ≥ 4 *possiede una corda*[4].

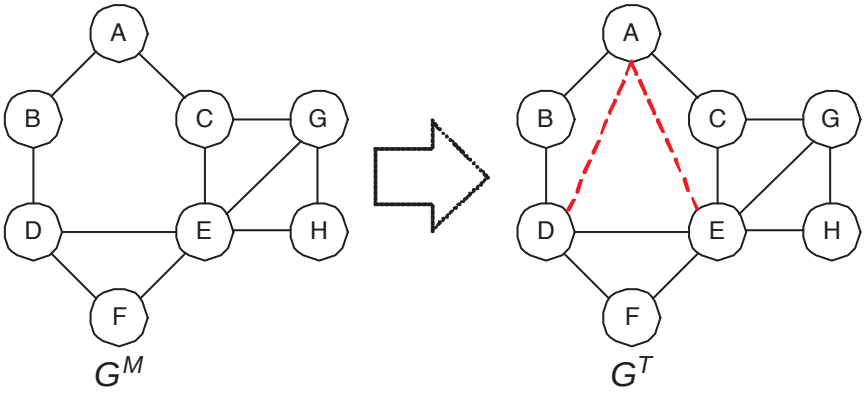


Figura 2.6. Esempio di grafo triangolato

Poichè le clique utilizzate per la costruzione del JT derivano direttamente dalle clique identificate nel grafo triangolato, è fondamentale trovare un modo efficiente per triangolare un grafo indiretto; Wen (1991) dimostra che la ricerca di una triangolarizzazione ottima è NP-hard. Si consideri il seguente:

Teorema 2.1 *(Jensen, 1996)*
Un grafo indiretto è triangolato se e solo se esiste una sequenza di eliminazione perfetta per i suoi nodi.

Una *sequenza di eliminazione perfetta* è un ordine secondo il quale si eliminano i nodi senza introdurre nel grafo archi di tipo *fill-ins*. Quest'ultimi sono definiti come nuovi archi da introdurre nel grafo per mantenere due nodi connessi. Se un grafo non è triangolato, è necessario introdurre *fill-ins*: in questo modo il grafo ottenuto ha una sequenza di eliminazione perfetta e risulta quindi essere triangolato.

Per determinare le clique di un grafo triangolato, si procede nel seguente modo (Jensen, 2001):

- si eliminano i nodi X_k tali che, detto N_{X_k} l'insieme dei nodi adiacenti a X_k e $F_{X_k} = N_{X_k} \cup \{X_k\}$, F_{X_k} è un sottografo completo;

[4] *Una corda in un ciclo di lunghezza* n *è un arco tra una coppia di vertici non adiacenti del ciclo.*

- esauriti i nodi del precedente tipo, si eliminano i nodi tali che la dimensione di F_{X_k} è minima, essendo la dimensione dell'insieme di nodi F_{X_k} definita come il prodotto del numero degli stati di ogni nodo che appartiene all'insieme; si introducono quindi *fill-ins*.

Si genera in questa maniera una sequenza di eliminazione perfetta e si determinano le clique F_{X_k} del grafo triangolato.

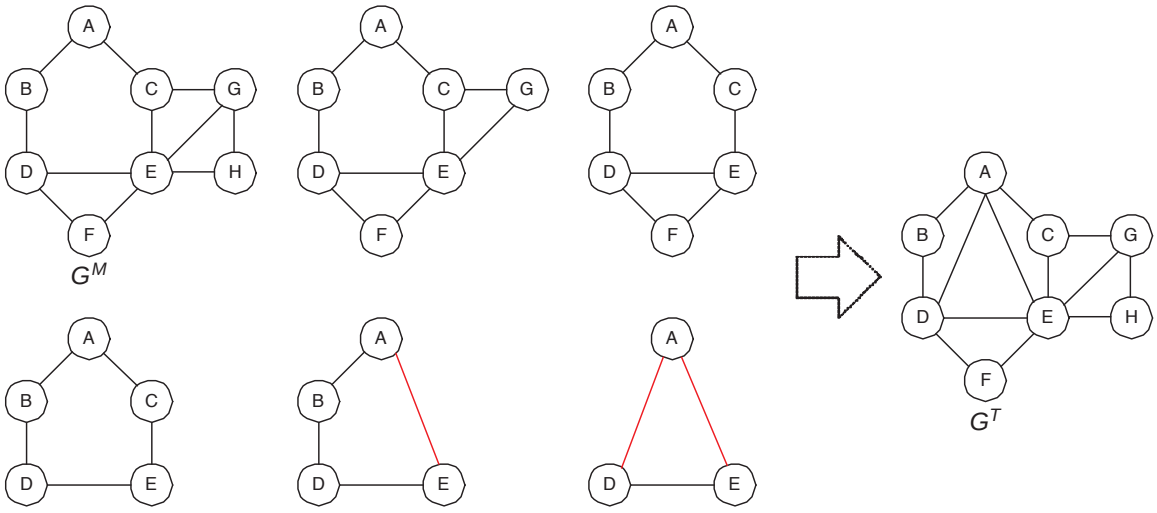


Figura 2.7. Triangolarizzazione di un grafo indiretto: determinazione della sequenza di eliminazione perfetta.

Proposizione 2.1
Tutte le sequenze di eliminazione perfetta producono lo stesso insieme di clique.

In Figura 2.7 si illustra il processo di triangolarizzazione di un grafo indiretto che segue l'ordine di eliminazione riportato in Tabella 2.1.

In generale, ci sono molti modi per triangolare un grafo indiretto. Il metodo introdotto, che si basa sulla minimizzazione della dimensione delle clique del grafo, produce una triangolarizzazione ottima (Golumbic, 1980; Kjærulff, 1990).

Le clique massimali indotte nella fase di triangolarizzazione diventano i vertici del JT. La determinazione del JT avviene collegando le clique in modo tale che l'albero risultante soddisfi le proprietà enunciate nella Definizione 2.1. Dato un insieme di n' clique, si può costruire un JT inserendo $n'-1$ archi, operazione che può essere tradotta nella determinazione di $n'-1$ separatori tra le varie coppie di clique (Jensen e Jensen, 1994).

vertice rimosso	clique indotta	fill-ins
H	EGH	-
G	CEG	-
F	DEF	-
C	ACE	A-E
B	ABD	A-D
D	ADE	-
E	AE	-
A	A	-

Tabella 2.1. Ordine di eliminazione

Esistono molti modi per costruire un JT; in particolare si ha che un JT è ottimale, se per ogni separatore determinato tra coppie di clique vale:

- la **massa**, definita come il numero di variabili che contiene ovvero il numero di variabili presenti nell'intersezione tra le due clique, è massima;
- il **costo**, definito come la somma dei pesi[5] delle due clique che separa, è minima.

La metodologia per la costruzione di un JT ottimale basato su questi principi si trova in Jensen e Jensen (1994). In Figura 2.8 si mostrano le clique e i separatori che determinano un JT ottimale, partendo dal grafo triangolato in Figura 2.6.

2.2.2 Inizializzazione e aggiornamento delle probabilità

Una volta costruito il JT, la componente quantitativa deve essere determinata in modo tale da soddisfare le proprietà di consistenza locale e globale. Ciò avviene:

- fissando i potenziali associati alle clique e ai separatori del JT, in modo che soddisfino alla consistenza globale. Il JT risultante è inconsistente poichè questa assegnazione iniziale dei potenziali non soddisfa anche alla condizione di consistenza locale;
- propagando i potenziali, attraverso *passaggi di informazione*, in modo che si arrivi a raggiungere la consistenza locale, ottenendo un JT consistente.

[5]Il *peso di una variabile* X_k è il numero di possibili stati di X_k. Il *peso di un insieme di variabili* $\mathbf{X}$ è il prodotto dei pesi delle variabili in $\mathbf{X}$.

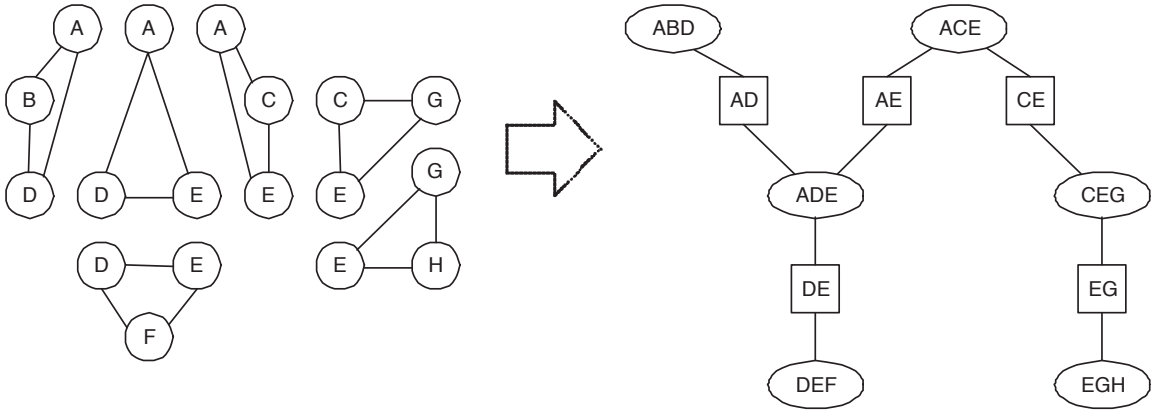


Figura 2.8. Esempio di JT ottimale.

La seguente procedura fissa i potenziali iniziali da assegnare al JT usando le probabilità condizionate specificate nella rete Bayesiana originaria:

1. Per ogni clique C_i e per ogni separatore S_j, porre tutti i potenziali uguali ad uno:

$$\phi_{C_i} = 1$$

$$\phi_{S_j} = 1$$

2. Per ogni variabile $X_k \in \mathbf{X}$, selezionare una clique C_i che contiene sia la variabile che tutti i suoi parenti $Pa(X_k)$ e moltiplicare il potenziale associato alla clique per $P(X_k|Pa(X_k))$:

$$\phi_{C_i} = \phi_{C_i} \cdot P(X_k|Pa(X_k))$$

Dopo aver fissato i potenziali iniziali, è soddisfatta la consistenza globale:

$$\frac{\prod_{i=1}^{n'} \phi_{C_i}}{\prod_{j=1}^{n'-1} \phi_{S_j}} = \frac{\prod_{k=1}^{n} P(X_k|Pa(X_k))}{1} = P(\mathbf{X})$$

dove n' rappresenta il numero totale delle clique, n il numero totale di variabili nel dominio, e ϕ_{C_i} e ϕ_{S_j} rispettivamente i potenziali delle clique e dei separatori del JT.

Perchè valga anche la consistenza locale, è necessario eseguire la **propagazione globale dei potenziali**. Quest'ultima è definita attraverso un insieme di *passaggi di informazione* tra due clique adiacenti.

Si considerino due clique C_1 e C_2 e il separatore adiacente ad entrambe S_0, a cui sono associati i rispettivi potenziali ϕ_{C_1}, ϕ_{C_2} e ϕ_{S_0}. Il passaggio di informazione tra C_1 e C_2 richiede due passi:

- Ottenere un nuovo potenziale per S_0 marginalizzando le variabili in C_1 che non appartengono a S_0:

$$\phi^{\star}_{S_0} = \sum_{C_1 \setminus S_0} \phi_{C_1}$$

- Ottenere un nuovo potenziale per C_2:

$$\phi^{\star}_{C_2} = \phi_{C_2} \lambda_{S_0}$$

dove

$$\lambda_{S_0} = \phi^{\star}_{S_0} / \phi_{S_0}$$

Per raggiungere la consistenza locale è necessario propagare i potenziali attraverso tutte le clique del JT. Dato un JT con n' clique, si seleziona una clique arbitraria C_0 e si eseguono $2(n'-1)$ passaggi di informazione, divisi in due fasi:

1. *COLLECT-EVIDENCE*: il passaggio dell'informazione avviene dalla periferia verso C_0;

2. *DISTRIBUTE-EVIDENCE*: il passaggio dell'informazione avviene da C_0 verso la periferia.

Ogni clique passa l'informazione, codificata attraverso il suo potenziale, a tutte le clique del JT in modo tale che il passaggio alla clique vicina avviene solo dopo avere ricevuto l'informazione da tutte le altre clique. Questa condizione assicura la consistenza locale del JT, una volta completata la propagazione globale (Jensen et al., 1990; Jensen, 2001).

La Figura 2.9 illustra il passaggio delle informazioni durante la propagazione globale sul JT rappresentato in Figura 2.8. La clique di partenza è quella evidenziata in grigio (contiene le variabili A, C ed E). Durante la fase COLLECT-EVIDENCE, l'informazione è passata in direzione della clique e avvengono i passaggi dall'1 al 5 indicati in Figura, mentre nella fase DISTRIBUTE-EVIDENCE l'informazione esce dalla clique iniziale seguendo i passaggi indicati in Figura dal 6 al 10. I numeri indicano uno dei possibili ordini di passaggio delle informazioni.

Alla fine dei passaggi precedenti, il JT risultate è consistente. Si può dunque calcolare $P(X_k)$ per qualsiasi variabile d'interesse X_k del dominio nel seguente modo:

1. Identificare un clique C_i che contiene X_k;

2. Calcolare $P(X_k)$ marginalizzando il potenziale ϕ_{C_i}:

$$P(X_k) = \sum_{C_i \setminus \{X_k\}} \phi_{C_i}.$$

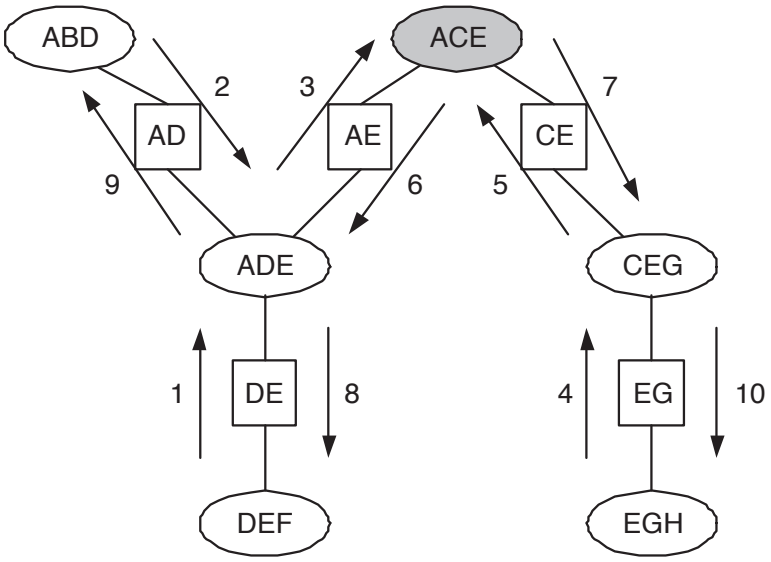


Figura 2.9. Passaggio delle informazioni durante la propagazione globale

Il processo appena descritto prende il nome di **inferenza senza evidenza**.

Nel caso in cui si dispone di informazione circa lo stato di una o di un insieme di variabili, *evidenza hard* denotata da **E**=**e**, è necessario modificare la procedura descritta in Sezione 2.2.2 per calcolare $P(X_k|\mathrm{e})$. Si introduce la funzione Λ_{X_k} a valori reali definita sullo spazio delle variabili X_k in modo tale che:

- Se $X_k \in \mathbf{E}$, ovvero si conosce lo stato assunto da X_k, allora:

$$\Lambda_{X_k}(x_k) = \begin{cases} 1, \text{ se } x_k \text{ è il valore assunto da } X_k \\ 0, \text{ altrimenti} \end{cases}$$

- Se $X_k \notin \mathbf{E}$, ovvero lo stato di X_k non è noto, allora:

$$\Lambda_{X_k}(x_k) = 1 \text{ per ogni valore } x_k.$$

Se non è presente evidenza la funzione Λ di ogni variabile consiste di tutti 1. Il processo di inizializzazione viene modificato integrando il calcolo della funzione Λ_{X_k} per ogni variabile X_k e incorporando l'evidenza che ne risulta nel JT attraverso la modificazione dei potenziali. In particolare, alle condizioni poste per i potenziali iniziali, si aggiunge la condizione che

$$\Lambda_{X_k} = 1$$

La funzione in questo caso codifica il fatto che non è ancora stata considerata alcuna evidenza.

Si incorpora ogni osservazione del tipo $X_k = x_k$ attraverso una nuova funzione $\lambda_{X_k}^{new}$, si identifica una clique C_i che contiene X_k e si aggiornano ϕ_{C_i} e Λ_{X_k} nel seguente modo:

$$\phi_{C_i} = \phi_{C_i} \Lambda_{X_k}^{new}$$

$$\Lambda_{X_k} = \Lambda_{X_k}^{new}.$$

Considerando l'evidenza $\mathbf{e}$, si modificano i potenziali del JT, in modo tale che tutte le probabilità che si calcolano attraverso esso, sono probabilità di eventi che sono legati all'evidenza $\mathbf{e}$. In altre parole, invece di calcolare $P(X_k)$ si calcola $P(X_k,\mathbf{e})$. Inoltre il JT, avendo modificato in tale maniera i suoi potenziali, codifica la probabilità $P(\mathbf{X},\mathbf{e})$ invece che $P(\mathbf{X})$.

Una volta effettuata la propagazione globale, il JT è consistente e per ogni clique C_i si ha il corrispondente potenziale $\phi_{C_i} = P(C_i|\mathbf{e})$. Marginalizzando per ottenere la probabilità di una variabile X_k si ottiene la probabilità congiunta di X_k ed $\mathbf{e}$:

$$P(X_k,\mathbf{e}) = \sum_{C_i \setminus \{X_k\}} \phi_{C_i}.$$

L'obiettivo è però il calcolo della probabilità di X_k *dato* $\mathbf{e}$, $P(X_k|\mathbf{e})$. Quest'ultima è ottenuta **normalizzando** $P(X_k,\mathbf{e})$ nel seguente modo:

$$P(X_k|\mathbf{e}) = \frac{P(X_k|\mathbf{e})}{P(\mathbf{e})} = \frac{P(X_k,\mathbf{e})}{\sum_{X_k} P(X_k,\mathbf{e})}.$$

La probabilità dell'evidenza $P(\mathbf{e})$ è spesso chiamata *costante di normalizzazione*.

Se dopo aver calcolato $P(X_k|\mathbf{e_1})$ si vuole calcolare $P(X_k\mathbf{e_2})$, dove $\mathbf{e_2}$ è un insieme diverso di evidenza, non è necessario ripetere tutto il processo di inizializzazione dei potenziali e di propagazione globale tra le clique. Huang e Darwiche (1994) descrivono in modo dettagliato come è possibile modificare i potenziali del JT, se l'evidenza viene modificata.

Capitolo 3

Apprendimento delle reti bayesiane

La scelta, o selezione del modello, è un aspetto importante e talvolta controverso, nell'ambito dell'analisi dei dati.

Nel contesto delle reti Bayesiane, la selezione del modello si identifica come il processo di ricerca del DAG, e i parametri associati, che meglio si adattano ai dati. Questo processo si traduce in termini di *apprendimento* della rete Bayesiana.

L'apprendimento di una rete Bayesiana si definisce dunque attraverso due fasi:

- Ricerca del grafo - *Apprendimento della struttura*;
- Stima delle probabilità condizionate - *Apprendimento dei parametri*.

Esistono tre metodi principali per l'apprendimento delle rete Bayesiane:

1. Modellare la conoscenza degli esperti, ovvero definire la struttura e i parametri sulla base delle valutazioni (soggettive) da parte di studiosi, esperti del problema oggetto di studio;
2. Indurre automaticamente la struttura dai dati;
3. Combinare i due metodi precedenti, usando la conoscenza dell'esperto come conoscenza a priori in modo tale da imporre la presenza o l'assenza di alcuni archi nella struttura o definire distribuzioni a priori sui parametri e/o sulla struttura.

Poichè il primo metodo risulta spesso costoso e non molto attendibile, oltre che temporalmente dispendioso, molta attenzione è rivolta allo studio delle tecniche di apprendimento automatico: ciò porta alla nascita di un numero sempre più crescente di algoritmi in grado di utilizzare l'informazione contenuta nei dati per specificare il modello, la rete Bayesiana, di riferimento in una analisi.

In questo Capitolo si presentano le tecniche più utilizzate per l'apprendimento delle reti Bayesiane; l'attenzione è rivolta principalmente al caso in cui sia la struttura che i parametri sono ignoti e i dati a disposizione sono completi[1].

3.1 Grafo noto: apprendimento dei parametri

Nel caso in cui tutte le variabili sono osservabili, l'approccio comunemente usato per stimare i parametri di una rete Bayesiana è quello della *stima di massima verosimiglianza* (MLE).

Assumendo che i dati seguano una distribuzione multinomiale e che valgano le ipotesi di indipendenza locale e globale per i parametri (Heckerman et. al, 1995), la verosimiglianza è scomponibile in accordo con la struttura della rete. In questo modo ogni parametro θ_{ijk} è stimato in modo indipendente e la sua stima si basa su statistiche sufficienti della forma N_{ijk}, dove N_{ijk} rappresenta il numero di casi nel database in cui la variabile X_i si trova nel suo stato k-esimo e l'insieme dei suoi parenti è complessivamente nello stato j-esimo. In particolare, si dimostra che le stime MLE sono del tipo:

$$_{MLE}\hat{\theta}_{ijk} = \frac{N_{ijk}}{N_{ij}}$$

dove $N_{ij} = \sum_k N_{ijk}$ L'uso delle tecniche Bayesiane per la stima dei parametri, porta a ricercare l'insieme dei parametri con massima probabilità a posteriori:

$$P(\Theta|D) \propto P(D|\Theta)P(\Theta) = L(D|\Theta)P(\Theta)$$

E' necessario dunque scegliere una priori per i parametri della rete, che comunemente è assunta coniugata rispetto alla distribuzione dei dati, in modo che la distribuzione a posteriori $P(\Theta|D)$ sia facilmente calcolabile. Ipotizzando per i dati l'uso di una distribuzione multinomiale, si assume come distribuzione a priori coniugata una Dirichlet, con iperparametro α_{ijk}. Con queste assunzioni, le stime Bayesiane dei parametri sono del tipo:

$$_{MAP}\hat{\theta}_{ijk} = \frac{\alpha_{ijk} + N_{ijk}}{\sum_k \alpha_{ijk} + N_{ij}}$$

In Naïm et al. (2004) è descritto il procedimento che porta alla specificazione delle espressioni di stima precedenti.

[1] Per dati completi si intende il caso in cui il database a disposizione non presenta dati mancanti.

3.2 Grafo non noto: apprendimento della struttura e dei parametri

Quando la struttura non è nota, l'apprendimento della rete Bayesiana avviene determinando il grafo e stimando, successivamente, i parametri corrispondenti.

Esistono due approcci principali:

- Approccio Search & Score: si usano funzioni *score* per confrontare la bontà delle possibili strutture della rete e si seleziona quella che si adatta meglio ai dati;
- Approccio Constraint-based: si utilizzano misure per scoprire le indipendenze condizionate tra le variabili e si seleziona la struttura che rappresenta al meglio queste relazioni.

Per l'obiettivo della tesi, decritto in Sezione 3.4, ci si focalizza sulle tecniche di apprendimento basate sull'approccio Search & Score. Quest'ultime risultano essere le tecniche maggiormente studiate e utilizzate in letteratura, essendo in grado di fornire risultati robusti[2] e non essendo sensibili ad errori nelle misure di indipendenza utilizzate nell'approccio Constraint-based.

3.2.1 Approccio Search & Score

L'obiettivo delle tecniche di apprendimento basate sull'approccio Search &Score, è quello di cercare la struttura che, usualmente, massimizza una funzione score, intesa come misura di adattamento tra il grafo e i dati. La ricerca avviene attraverso un processo di esplorazione nello spazio dei possibili DAG, calcolando iterativamente la funzione score in modo da valutare l'adattamento di ogni struttura candidata e selezionando quella a cui corrisponde valore più alto. Ogni algoritmo è caratterizzato da una specifica funzione score e da una procedura che definisce lo spazio di ricerca.

Le funzioni score possono essere definite in base a differenti approcci. A seguito, vengono descritti i principali metodi utilizzati in letteratura.

Essendo le funzioni score intese come misura di adattamento della struttura ai dati, la *Log-verosimiglianza* risulta essere indicata per l'obiettivo specificato. Essa ha il vantaggio di essere calcolata in modo efficiente: la Log-verosimiglianza gode della proprietà di scomponibililità in fattori, detti funzioni score locali, in accordo con la struttura considerata. Questa

[2] Apprendere reti Bayesiane robuste significa apprendere reti per le quali piccole alterazioni nel modello non influenzano in modo drastico il comportamento del sistema (Cozman, 1996)

decomposizione locale permette una valutazione rapida della variazione della score tra due strutture, in funzione di un numero ridotto di score locali. Si ha che:

$$\begin{aligned} Score_{ML}(G,D) &= \log L(G,D) \\ &= \log P(D|G) \\ &= \log \prod_{i=1}^{n} \prod_{j=1}^{q_i} \prod_{k=1}^{r_i} \theta_{ijk}^{N_{ijk}} \\ &= \sum_{i=1}^{n} \sum_{j=1}^{q_i} \sum_{k=1}^{r_i} N_{ijk} \log \frac{N_{ijk}}{N_{ij}} \end{aligned}$$

dove n rappresenta il numero di variabili del dominio, r_i il numero di stati che può assumere la variabile X_i e q_i il numero delle possibili combinazioni degli stati di $Pa(X_i)$.

L'uso della Log-verosimiglianza non implica nessun vincolo sulla complessità della struttura ricercata: per un database D fissato, la struttura più verosimile sarà quella che possiede il numero più grande di parametri (Friedman et al., 1997) , ovvero la struttura che lega tutte le variabili (strutture di questo tipo vengono dette *grafi completi*).

La maggior parte delle funzioni score di uso corrente applicano il principio di parsimonia del *rasoio di Occam*: trovare il modello che corrisponde meglio ai dati D e che risulti il più semplice possibile; in particolare le funzioni score basate sull'*entropia* applicano questo principio. Esse sono definite attraverso due componenti: l'entropia condizionata della struttura e la complessità del modello, intesa come il numero di parametri necessario per specificare la rete Bayesiana.

L'entropia condizionata della struttura[3] è definita da:

$$H(G,D) = \sum_{i=1}^{n} \sum_{j=1}^{q_i} \sum_{k=1}^{r_i} -\frac{N_{ijk}}{N} \log \frac{N_{ijk}}{N_{ij}}.$$

dove con N si indica la numerosità dei casi del database in esame, mentre la complessità della rete da:

$$K = \sum_{i=1}^{n} (r_i - 1) q_i.$$

A partire dalla definizione generale di funzione score basata sull'entropia, le misure più frequentemente utilizzate in letteratura per l'apprendimento delle reti Bayesiane sono:

- La score AIC (Akaike, 1970):

$$Score_{AIC} = -N \cdot H(G,D) - K$$

- La score BIC (Schwartz, 1978):

$$Score_{BIC} = -N \cdot H(G,D) - \frac{K}{2} \log N$$

[3] La Log-Verosimiglianza e l'entropia condizionata sono legate da una legge diretta tale che $\log L(D|G) = -N \cdot H(G,D)$.

In letteratura viene inoltre proposta la funzione score *MDL* (Rissanen, 1978), basata sul principio di *Minimum Description Length*. Esso afferma che il modello che rappresenta meglio un insieme di dati è quello che minimizza la somma di due termini: la lunghezza del codice del modello e la lunghezza del codice dei dati, entrambi misurati in bits. Lam e Bacchus (1994) dimostrano che questa funzione score è definita come:

$$Score_{MDL} = -\log P(D|G) + \frac{K}{2}\log N.$$

e risulta uguale all'opposto della score BIC. Quindi, minimizzare la score MDL nella selezione del modello porta agli stessi risultati che si otterrebbero massimizzando la score BIC.

L'approccio Bayesiano è un metodo pratico e ben definito. L'idea base in questo contesto è di usare come funzione score la probabilità a posteriori della struttura della rete, dato il database. Attraverso una legge a priori sulle strutture $P(G)$[4], si esprime la probabilità a posteriori delle strutture dato il database $P(G|D)$:

$$\begin{aligned} Score(G,D) &= P(G|D) \\ &= \tfrac{P(D|G)P(G)}{P(D)} \\ &\propto P(D|G)P(G) \end{aligned}$$

Con le usuali ipotesi di indipendenza dei parametri (Sezione 3.1), e supponendo che essi si distribuiscano secondo una legge a priori Dirichlet, si dimostra che la score Bayesiana è definita nel seguente modo:

$$Score_{Bayes} = P(G)\prod_{i=1}^{n}\prod_{j=1}^{q_i}\frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij}+N_{ij})}\prod_{k=1}^{r_i}\frac{\Gamma(\alpha_{ijk}+N_{ijk})}{\Gamma(\alpha_{ijk})}$$

dove Γ rappresenta la funzione *Gamma*.

Cooper e Heskovits (1992) pongono $\alpha_{ijk} = 1$ ($\alpha_{ij} = r_i$), ottenendo la score $K2$; Heckerman et al. (1995) modificano la score fissando $\alpha_{ijk} = \frac{1}{r_i}\cdot q_i$ ($\alpha_{ij} = \frac{1}{q_i}$), definendo la score BDe che ha la proprietà aggiuntiva di essere uguale per le strutture appartenenti alla stessa classe di equivalenza. Per un confronto e una maggiore caratterizzazione delle funzioni score introdotte, si rimanda a Bouckaert (1995).

Idealmente si vorrebbe calcolare la funzione score selezionata per tutte le strutture costruibili sull'insieme di variabili considerato. Robinson (1977) prova che il numero $NG(n)$ di diverse strutture identificabili a partire da n nodi è dato dalla seguente formula ricorsiva:

$$NG(n) = \begin{cases} 1 & n = 0{,}1 \\ \sum_{i=1}^{n}(-1)^{i+1}\binom{n}{i}2^{i(n-1)}NG(n-i) & n > 1 \end{cases}$$

[4]Se non si hanno informazioni aggiuntive, si assume per la struttura una distribuzioni a priori uniforme.

n	$NG(n)$
0	1
1	1
2	3
3	25
4	543
5	29281
6	3781503
7	1138779265
8	783702329343

Tabella 3.1. Numero di possibili DAG con n nodi.

In Tabella 3.1 si calcola $NG(n)$ per valori piccoli di n. Poichè il numero di strutture possibili cresce esponenzialmente con il numero di nodi, non è computazionalmente fattibile calcolare la funzione score per tutte le possibili strutture. Chickering (1996) dimostra che trovare la struttura a cui corrisponde la funzione score più alta, nel caso in cui ogni nodo ha al massimo K parenti, è NP-hard per $K > 1$.

In letteratura vengono perciò proposte procedure di ricerca euristiche; la procedura più comunemente usata è l'algoritmo greedy Hill-Climbing (Cooper e Herskovits, 1992; Buntine, 1991). Utilizzando questo metodo, la ricerca inizia da un grafo vuoto, e determinando tra i possibili cambi locali (aggiunta, cancellazione e cambio della direzione di un arco) quello, se esiste, che incrementa maggiormente la funzione score. La modifica locale selezionata è considerata come punto di partenza per l'iterazione successiva. La procedura si arresta quando nessun cambio locale aggiuntivo aumenta la funzione score e la struttura finale presenta l'insieme delle modifiche locali selezionate durante l'intera procedura.

Non c'è garanzia che l'algoritmo produca un massimo globale: per evitare il problema si utilizzano comunemente teniche di Random Restart o Simulated Annealing (Bouckaert, 1995).

In letteratura sono proposte, oltre ai metodi Hill-Climbing, altre procedure di ricerca euristica come algoritmi di tipo Best-first (Russel e Norvig, 1995) o algoritmi genetici (Larrañaga et al., 1996).

Esistono, inoltre, metodologie che permettono di ridurre lo spazio di ricerca (lo spazio dei DAG) in sottospazi particolari, come ad esempio lo spazio degli alberi, che utilizza la

Algoritmo K2

1. Per $i = 1$ fino a n:
   (a) $Pa(X_i) = \emptyset$; $OK = vero$
   (b) $Score_{old} = score(X_i, Pa(X_i))$
   (c) Finchè $OK = vero$ e $Pa(X_i) < q_i$
      i. Sia z il nodo nell'insieme dei precedessori di X_i che non appartiene a $Pa(X_i)$ che massimiza $score(X_i, Pa(X_i) \cup \{z\})$
      ii. $Score_{new} = score(X_i, Pa(X_i) \cup \{z\})$
      iii. Se $Score_{new} > Score_{old}$ allora
      $Score_{old} > Score_{new}$ e $Pa(X_i) = Pa(X_i) \cup \{z\}$
      Altrimenti $OK = falso$
   (d) I parenti del nodo X_i sono $Pa(X_i)$

Figura 3.1. Algoritmo K2

nozione di *Maximum Weight Spanning Tree* (Chow e Liu, 1968b; Heckerman et al., 1995), lo spazio delle classi di equivalenza delle strutture (Meek, 1997; Chickering, 2002) o lo spazio dell'ordinamento dei nodi (Cooper e Herskovits, 1992).

L'algoritmo $K2$ di Cooper e Hersovits (1992) si basa su quest'ultimo principio, e utilizza come funzione score la metrica K2 precedentemente definita. L'algoritmo assume un ordinamento sull'insieme delle variabili $\{X_1,X_2,\ldots,X_n\}$ tale che X_j non può essere parente di X_i se $j > i$, limitando il numero di possibili strutture a $NG'(n) = 2^{n(n-1)/2}$; supponendo di avere un insieme ordinato di $n = 3$ variabili, $NG'(3) = 8$ contro $NG(3) = 25$.

In Figura 3.1 è descritto il pseudocodice dell'algoritmo, mentre in Figura 3.2 vengono mostrati i passi dell'algoritmo, supponendo $n = 3$ e assumendo per le variabili l'ordinamento $\{X,Y,Z\}$.

Lo svantaggio maggiore dell'algoritmo K2 è che l'ordinamento richiesto sui nodi influenza la rete risultante. Per garantire un buon comportamento dell'algoritmo è essenziale quindi scegliere un 'buon ordinamento', che può essere fornito da un esperto o può essere ricercato attraverso l'utilizzo di approcci di tipo algoritmi genetici (Larrañaga et al., 1996; Hsu et al., 2002).

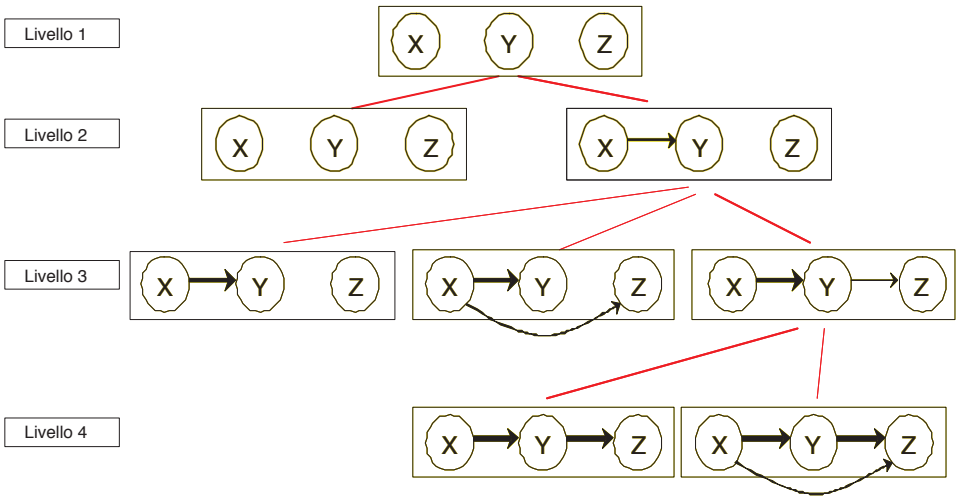


Figura 3.2. Esempio di esecuzione dell'algoritmo K2

Puerta Callejon (2001) propone una modificazione dell'algoritmo K2, l'algoritmo K2SN, che identifica un'ordine, in fase d'opera, tra le variabili e costruisce, utilizzando l'algoritmo K2, una rete compatibile con l'ordine trovato. In Figura 3.3 è descritto il pseudocodice dell'algoritmo.

3.2.2 Approccio Constraint-based

Gli algoritmi di tipo Constraint-based si sviluppano nel seguente modo:

- Studio delle relazioni tra le variabili attraverso misure di indipendenza;
- Determinazione della struttura che rappresenta il più possibile queste relazioni.

Le misure di indipendenza tipicamente usate sono i test statistici χ^2 e il rapporto di verosimiglianza G^2. In generale, si assume:

- *Sufficienza casuale*: non ci sono variabili non osservate (hidden o latenti) che sono parenti di una o più variabili osservate nel dominio;
- *Proprietà di Markov*: una variabile è condizionatamente indipendente dai suoi non discendenti dati i suoi parenti;
- *Faithfulness*: un grafo G e una distribuzione di probabilità P sono Faithful se tutte e sole le relazioni di indipendenza valide in P sono quelle implicate dalle proprietà di Markov in G.

Algoritmo K2SN

1. $Analysed = \emptyset$ e $ToAnalyse = \{X_1, \ldots, X_n\}$
2. $max = -\infty$
3. Per $i = 1$ fino a n
   Se $score(X_i, \emptyset) > max$ allora
   $max = score(X_i, \emptyset)$ e $X = X_i$
4. $Analysed = Analysed \cup \{X\}$
   $ToAnalyse = ToAnalyse \setminus \{X\}$
   $Pa(X) = \emptyset$
5. Finchè $ToAnalysed \neq \emptyset$
   (a) $max = \infty$
   (b) Per $i = 1$ fino a n
      i. Sia $X_i \in ToAnalyse$
      ii. $score = K2(X_1, Analysed)$
      iii. $Pa(X_i) = K2(X_i, Analysed)$
      iv. Se $score(X_i, Pa(X_i)) > max$ allora
      $max = score(X_i, Pa(X_i))$ e $X = X_i$ e $Pa(X) = Pa(X_i)$
   (c) $Analysed = Analysed \cup \{X\}$
      $ToAnalyse = ToAnalyse \setminus \{X\}$

Figura 3.3. Algoritmo K2SN

E'dunque possibile determinare un insieme di vincoli sulla struttura di una rete Bayesiana: una indipendenza tra due variabili si traduce attraverso l'assenza di un arco tra i nodi che le rappresentano, una dipendenza condizionata corrisponde a una V-struttura, etc...

Basandosi su questi principi, Spirtes et al. (1993) propongono l'algoritmo SGS: partendo da un grafo non orientato totalmente connesso, si calcolano test di indipendenza condizionata per saggiare l'esistenza di un arco tra due variabili. Successivamente si determina, se possibile, la direzionalità degli archi individuati nella fase precedente, esaminando tutte le V-strutture presenti fra le variabili, mantenendo l'aciclicità del grafo. Questo metodo richiede un numero di test esponenziale in rapporto al numero di variabili.

Spirtes et al. (1993) definiscono successivamente una variante dell'algoritmo SGS, l'algoritmo PC, che limita il numero di test di indipendenza condizionata, ordinandoli, da un ordine minore (indipendenza condizionata di ordine zero che corrisponde ad un test di indipendenza semplice tra due variabili) a un ordine maggiore.

L'algoritmo IC, proposto da Pearl (2000) è basato sullo stesso principio, ma determina un grafo non orientato aggiungendo legami invece di eliminarli. Pearl e Verma (1991) avevano già in precedenza proposto una versione dell'algoritmo IC, che prendeva in considerazione le variabili latenti, rinominato IC* in Pearl (2000).

In letteratura esistono altre proposte di metodologie per l'apprendimento della struttura della reti Bayesiane che si basano sull'approccio Constraint-based.

L'algoritmo BN-PC (Cheng et al., 1997) usa il concetto di mutua informazione[5] e richiede un ordinamento delle variabili. La mutua informazione come misura di indipendenza statistica può essere considerata come un coefficiente di correlazione generalizzato, rispetto alla natura della relazione fra le variabili, perciò non soltanto lineare, come l'usuale coefficiente di correlazione.

L'algoritmo GS (Margaritis, 2003) utilizza il Markov Blanket di una variabile: l'uso di questa nozione rende più semplice la valutazione e l'apprendimento in un contesto semi-automatico in cui è disponibile informazione a priori di un esperto in grado di verificare la veridicità dei risultati dei test di indipendenza coinvolti.

3.3 Dati incompleti

In Sezione precedente si sono presentate le metodologie di apprendimento dei parametri e della struttura di una rete Bayesiana nel caso di dati completi. Quando il database analizzato

[5]Nella Teoria dell'Informazione, la mutua informazione tra due variabili X e Y è definita come $MI(X,Y) = \sum_{x,y} P(X=x,Y=y) \log \frac{P(X=x,Y=y)}{P(X=x)P(Y=y)}$.

presenta valori mancanti, le tecniche descritte risentono del fatto che alcune caratterizzazioni o assunzioni non sono più valide.

In seguito vengono presentati sinteticamente gli approcci più utilizzati in letteratura per l'apprendimento dei parametri nel caso di struttura nota e non nota.

Quando il database presenta dati mancanti, la verosimiglianza non risulta più scomponibile in accordo con la struttura. Per calcolare le stime MLE dei parametri della rete, si utilizza comunemente l'algoritmo iterativo EM (Dempster et al., 1977; McLachlan e Krishnan, 1997). Questo algoritmo, assume che il meccanismo generatore dei dati mancanti sia del tipo *Missing At Random* (MAR) . La stessa assunzione viene fatta per la ricerca di stime dei parametri della rete utilizzando tecniche di tipo Monte Carlo, come l'algoritmo Gibbs Sampling (Chib, 1995; Chickering e Heckerman, 1997).

Altre metodologie sono state proposte per la stima dei parametri in caso di dati incompleti, che non si basano sull'assunzione MAR; si veda ad esempio Cooper (1995a), Spirtes et al. (1995), Ramoni e Sebastiani (1998).

Nel caso di struttura ignota, l'utilizzo di approcci di tipo Search & Score, basati sulla Log-verosimiglianza, presenta lo stesso problema descritto nel caso di struttura nota.

La metodologia dell'algoritmo EM, per la stima dei parametri della rete, può essere utilizzata anche nel caso in cui la struttura non sia fissata. Friedman (1998) propone l'algoritmo SEM - Structural EM -, in cui si effettua una ricerca iterativa congiunta nello spazio dei parametri e in quello delle strutture. Analogamente, Ramoni e Sebastiani (1997) modificano l'algoritmo BC - Bound and Collapse - proposto come metodo per stimare probabilità condizionate nel caso di database incompleti, proponendo un metodo deterministico in grado di apprendere sia la struttura che i parametri della rete Bayesiana.

L'approccio Constraint-based assume l'ipotesi di sufficienza casuale. Se il database presenta dati mancanti, questa assunzione non è più valida.

In letteratura esistono metodologie, che risultano essere modifiche degli algoritmi proposti nel caso di dati completi, in grado di trattare questo problema: l'algoritmo FCI (*Fast Causal Inference*) proposto da Glymour e Cooper (1999) e l'algoritmo IC* di Pearl (2000). Come per PC e IC, la differenza principale nei due algoritmi è la costruzione del grafo non orientato di partenza: eliminazione degli archi a partire da un grafo totalmente connesso per FCI e aggiunta degli archi partendo da un grafo vuoto per IC*.

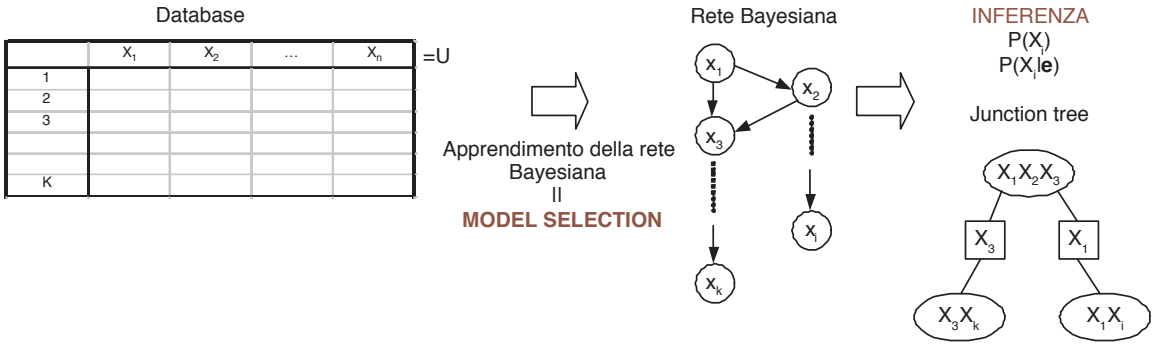


Figura 3.4. Processo classico di selezione del modello

3.4 Approccio per la selezione del modello basato sui Junction Tree

La selezione del modello, intesa come il processo di apprendimento della rete Bayesiana, si sviluppa a partire da un database di osservazioni sulle variabili del dominio oggetto di studio attraverso l'applicazione di uno specifico algoritmo di apprendimento. Il risultato è la determinazione di una rete Bayesiana (struttura e parametri del modello) sulla quale si sviluppa il processo di inferenza probabilistica attraverso l'uso di tecniche quale la Junction Tree Propagation (Figura 3.4).

Le funzioni score comunemente utilizzate nell'apprendimento della struttura della rete basate sull'approccio Search & Score, come ad esempio la Score BIC, AIC o MDL, sono spesso definite come un compromesso tra l'adattamento del modello ai dati (Log-verosimiglianza) e la complessità del grafo

$$\begin{aligned} Score_{ENTROPIA} &= -N \cdot H(G,D) - f(N) \\ &= -N \cdot \sum_{i=1}^{n} \sum_{j=1}^{q_i} \sum_{k=1}^{r_i} -\frac{N_{ijk}}{N} \log \frac{N_{ijk}}{N_{ij}} - f(N) \cdot K \\ &= \log L(D|G) - f(N) \cdot K. \end{aligned}$$

.

La complessità di rappresentazione della rete K, definita in Sezione 3.2, rappresenta il numero di parametri indipendenti, ovvero il numero minimo di probabilità condizionate e a priori necessarie per specificare la rete.

Le funzioni score sono dunque indipendenti da altre caratterizzazioni, quale ad esempio la complessità del JT associato alla rete appresa. L'inferenza probabilistica descritta nel Capitolo 2 presenta una complessità, o *costo*, legata alla metodologia utilizzata nel calcolo.

Considerando l'approccio basato sulla Junction Tree Propagation, la complessità inferenziale è determinata principalmente dalla dimensione dello spazio delle clique del JT.

Supponendo che la rete sia stata trasformata in un JT con n' clique C_1,C_2,...,$C_{n'}$, una misura della dimensione dello spazio delle clique è definita attraverso la **complessità del JT**: la somma, per tutte le clique, del prodotto del numero di stati delle variabili presenti in ogni clique

$$K_{JT} = \sum_{C_i \in \{C_1,\ldots,C_{n'}\}} \prod_{X \in C_i} |X| \tag{3.1}$$

dove $|X|$ è il prodotto degli stati della variabile X. Utilizzando questa quantità è possibile, sia confrontare i JT ottenuti da differenti triangolarizzazioni, sia le reti, in modo da individuare quella con complessità inferenziale minore.

E' dimostrato (Beygelzimer e Rish, 2002) che due reti che si adattano ugualmente bene ai dati e che presentano complessità di rappresentazione simile, possono avere complessità inferenziale abbastanza diverse: complessità di rappresentazioni piccole non implicano complessità del JT piccole .

Se la rete Bayesiana ha come scopo l'inferenza sulle variabili del dominio, la funzione score, determinante nel processo di selezione del modello, dovrebbe tenere in considerazione la complessità del JT.

La funzione score proposta risulta essere un compromesso tra la verosimiglianza del modello appreso e la complessità del processo di inferenza probabilistica, misurato attraverso la complessità del JT.

Specificatamente, essa è:

$$\begin{aligned} Score_{JT} &= -N \cdot H(G,D) - f(N,n) \cdot K_{JT} \\ &= \log L(D|G) - f(N,n) \cdot K_{JT} \end{aligned}$$

con $f(N,n)$ funzione reale non negativa definita nel seguente modo:

$$f(N,n) = \frac{1}{2} \cdot n \cdot \log N$$

dove n è il numero di variabili e N la numerosità del database.

Nishii (1988) dimostra che per avere una score basata sull'entropia consistente devono valere le seguenti condizioni:

$$\lim_{N\to\infty} f(N) = \infty \qquad \text{e} \qquad \lim_{N\to\infty} \frac{f(N)}{N} = 0.$$

che, nel caso considerato, si indentificano in:

$$\lim_{\substack{n\to\infty \\ N\to\infty}} f(n,N) = \infty \qquad \text{e} \qquad \lim_{\substack{n\to\infty \\ N\to\infty}} \frac{f(n,N)}{n \cdot N} = 0.$$

Per come è definita la funzione $f(n,N)$, queste condizioni sono soddisfatte.

Calcolare la funzione score ad ogni passo successivo della procedura di ricerca utilizzata, può rendere il processo di selezione del modello intrattabile per il numero elevato di calcoli che devono essere eseguiti. Per questo motivo le funzioni score, prevalentemente usate in letteratura, sono scomponibili ovvero risultano essere la somma di funzioni score locali legate alle singole variabili della rete:

$$Score(G,D) = \sum_{i=1}^{n} score(X_i,Pa(X_i)).$$

Ciò permette di valutare la variazione della funzione score tra due strutture considerando solo il sottografo a cui appartiene il cambio locale effettuato: l'aggiunta, ad esempio, di un parente ad una variabile del dominio in grado di incrementare la funzione score locale legata a questa variabile, per la proprietà di scomponibilità porterà ad un aumento anche del valore della funzione score globale, portando quindi alla determinazione dell'arco corrispondente nella struttura della rete.

La funzione score $Score_{JT}$ definita precedentemente non gode di questa proprietà, e il suo utilizzo durante la procedura di ricerca porterebbe al calcolo, ad ogni passo successivo, della complessità del JT corrispondente alla struttura in esame.

Per agire a livello locale si definisce una misura, detta **JT-Based**, che risulta essere un compromesso tra la verosimiglianza del cambio locale considerato e la complessità del JT corrispondente al sottografo che lo contiene,

Si definisce l'euristica:

$$score_{JT-Based}(X_i,Pa(X_i)) = \sum_{j}\sum_{k} N_{ijk} \log \frac{N_{ijk}}{N_{ij}} - \frac{1}{2} \cdot n_i \cdot \log N \cdot K_{JT}(X_i,Pa(X_i)) \quad (3.2)$$

dove n_i è il numero di variabili nell'insieme $(X_i \cup Pa(X_i))$ e $K_{JT}(X_i,Pa(X_i))$ è la complessità del JT, definita come in Equazione 3.1, calcolata considerando solo il sottografo limitato a queste variabili. Il primo termine dell'espressione a secondo membro rappresenta la verosimiglianza associata a $(X_i,Pa(X_i))$.

In Figura 3.5 si presenta un esempio di procedura di ricerca K2SN, descritta nella Sezione 3.2 con l'utilizzo della euristica locale JT-Based proposta.

L'esempio considera tre variabili $\{X_1,X_2,X_3\}$, ripettivamente con 2, 3 e 4 possibili stati assunti.

Livello I.

L'algoritmo cerca il nodo che, tra tutti, massimizza la misura locale, definita dall'Equazione 3.2, assumendo vuoto l'insieme dei parenti: $score_{JT-Based}(X_i,\emptyset)$. Per definizione

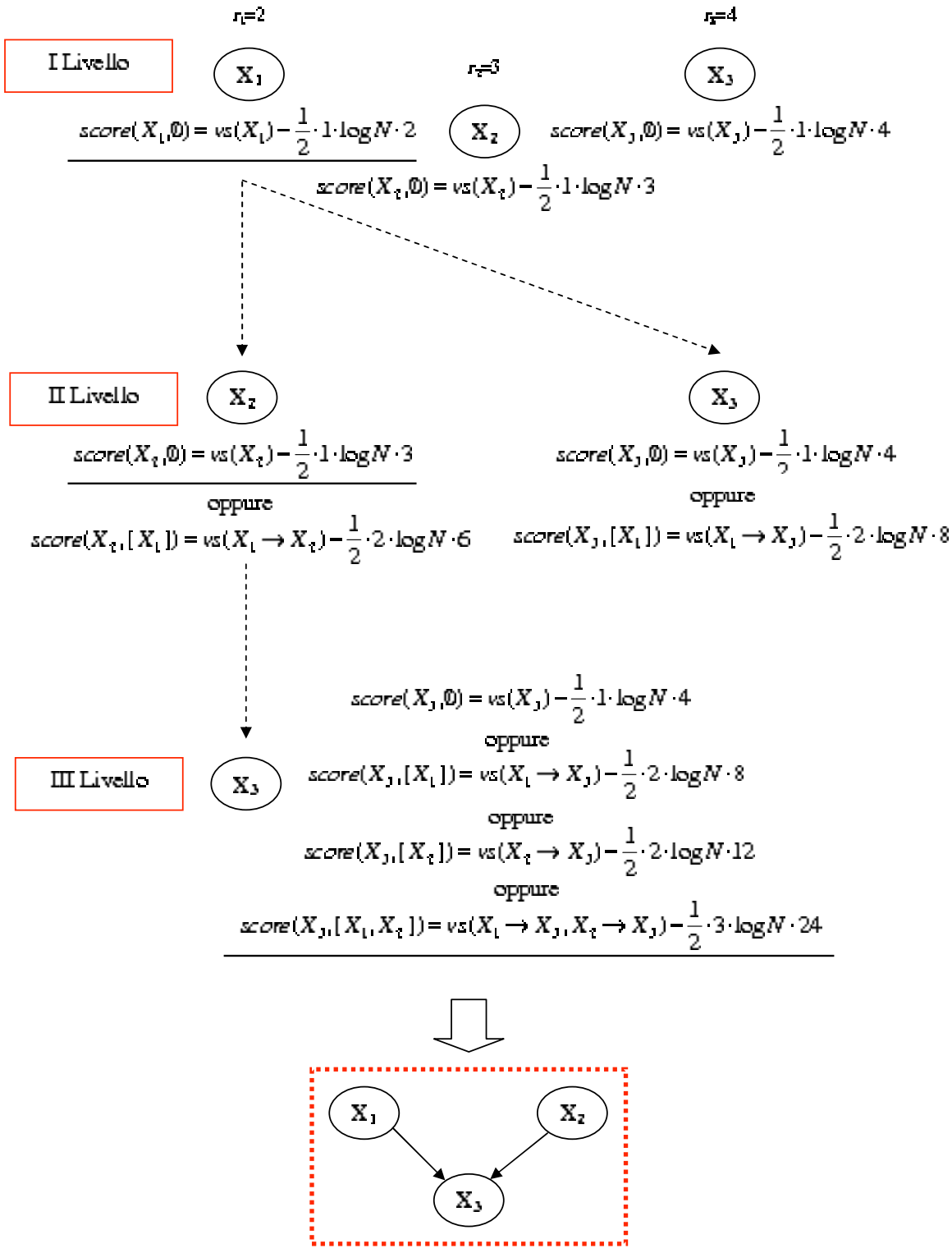


Figura 3.5. Esempio di procedura di ricerca con l'utilizzo della funzione score locale JT-Based

quest'ultima risulta:

$$score_{JT-Based}(X_i,) = \sum_k N_{ik} \log \frac{N_{ik}}{N_i} - \frac{1}{2} \cdot 1 \cdot \log N \cdot K_{JT}(X_i,\emptyset)$$

dove $N_{ik} = \sum_j N_{ijk}$, $N_i = \sum_j N_{ij}$ e $K_{JT}(X_i,\emptyset)$ è la complessità del JT associato al sottografo che considera solo la variabile X_i (il JT formato solo da una clique contenente la variabile) misurata attraverso il numero di possibili stati della variabile.

Livello II.

Si cerca quale fra i rimanenti nodi massimizza la misura locale, supponendo che i nodi trovati precedentemente (in questo caso il nodo trovato a Livello I) possano far parte dell'insieme dei parenti. Per valutare questo massimo, si inizia con un insieme dei parenti vuoto e si procede inserendo i nodi nell'insieme dei parenti che incrementano il valore della misura considerata, fino a che non si riscontrano ulteriori miglioramenti, e cosi per tutti i nodi.

In Figura 3.5, avendo determinato che la variabile X_1 presenta funzione score locale di valore maggiore, a *Livello II*, le funzioni score locali confrontate sono calcolate per le variabili X_2 e X_3 e con possibili parenti l'insieme vuoto o X_1. Nel primo caso la funzione score locale è definita come a Livello 1, mentre nel secondo caso si ha:

$$score_{JT-Based}(X_i,X_1) = \sum_k N_{ijk} \log \frac{N_{ijk}}{N_{ij}} - \frac{1}{2} \cdot 2 \cdot \log N \cdot K_{JT}(X_i,X_1)$$

dove N_{ijk} e N_{ij} sono definite nel modo usale ($i = \{2,3\}$ è l'indice della variabile, j è il numero delle possibili configurazioni dell'insieme dei parenti di X_i che in questo caso corrisponde a X_1 ed è dunque pari a 2, e k è il numero dei possibili stati assunti dalla variabile X_i), mentre $K_{JT}(X_i,X_1)$ è la complessità del JT associato al sottografo che considera solo le variabili $\{X_i,X_1\}$ e X_1 è parente di X_i.

Il JT corrispondente a questo sottografo è formato da una sola clique che contiene entrambe le variabili e la sua complessità risulta essere il prodotto dei possibili stati delle variabili (pari a 6 nel caso si considerino le variabili $\{X_2,X_1\}$, pari a 8 se si considerano le variabili $\{X_3,X_1\}$).

Livello III.

Dato che nell'esempio si assume che la funzione score locale più alta associata alla variabile X_2 corrisponda ad un insieme dei parenti vuoto, si passa ad esaminare le restanti variabili, ovvero, nell'esempio, la variabile X_3. Si confrontano i valori delle funzioni score locali considerando come insieme dei parenti per X_3 l'insieme vuoto, l'insieme determinato da una variabile tra X_1 e X_2 e l'insieme determinato da entrambe le variabili X_1 e X_2.

Solo l'ultima funzione score locale non è stata ancora definita, essendo state le altre calcolate nei livelli precedenti, e corrisponde a:

$$score_{JT-Based}(X_3,\{X_1,X_2\}) = \sum_k N_{ijk} \log \frac{N_{ijk}}{N_{ij}} - \frac{1}{2} \cdot 3 \cdot \log N \cdot K_{JT}(X_3,\{X_1,X_2\})$$

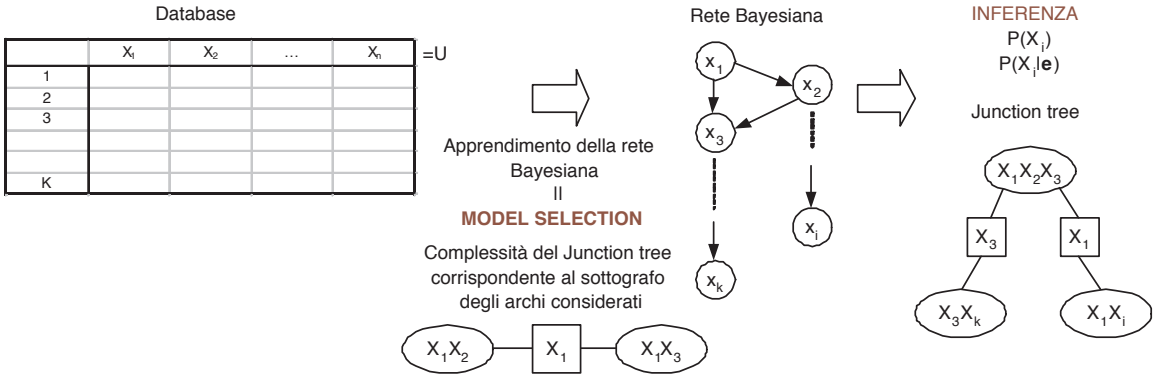


Figura 3.6. Processo di selezione del modello basato sulla complessità del JT

dove N_{ijk} e N_{ij} sono definite come sopra e nel caso in cui l'insieme dei parenti è formato dalle due variabili $\{X_1,X_2\}$, mentre $K_{JT}(X_3,\{X_1,X_2\})$ è la complessità del JT associato al sottografo che considera le variabili $\{X_1,X_2,X_3\}$ in modo che X_1 e X_2 siano entrambe parenti di X_3. Il JT corrispondente a questo sottografo è formato da una sola clique che contiene tutte le variabili (per il processo di moralizzazione, essendo X_1 e X_2 parenti della stessa variabile) e la sua complessità è pari a 24, prodotto dei possibili stati delle variabili.

Anche se, in quest'ultimo caso, la complessità del JT risultante è maggiore, la Log-verosimiglianza dei legami valutati è tale da bilanciare questo aumento; se la Log-verosimglianza è tale da superare in valore la complessità risultate del JT, corrispondente al sottografo che comprende l'arco o l'insieme degli archi valutati, allora questo arco o insieme di archi verrà aggiunto alla struttura rete.

In Figura 3.5 si ipotizza che il valore della funzione score locale più alta sia quella sottolineata.

Poichè a Livelli successivi, si considerano valori già calcolati in precedenza, vengono utilizzate tecniche che permettono di memorizzare i calcoli effettuati evitando il ricalcolo di queste quantità, se richieste, durante la procedura di ricerca.

L'implementazione della procedura locale proposta è stata implemetata in JAVA utilizzando i codici sorgente API del software **Elvira**. Elvira, scaricabile gratuitamente da Internet (http://www.ual.es/personal/asalmero/elvira/presenta01.swf), presenta un ampio insieme di algoritmi di apprendimento e di inferenza (sia esatta che approssimata) e utili strumenti di analisi nel contesto delle reti Bayesiane. Il codice JAVA è stato implementato nell'ambiente di sviluppo Java open source *Eclipse* (http://www.eclipse.org/), versione per Windows (il sistema operativo del pc utilizzato è Windows XP Professional, versione 2002).

La Figura 3.6 descrive il nuovo approccio proposto per la selezione del modello . A differenza del processo classico rappresentato in Figura 3.4, la determinazione del JT avviene nella fase precedente la selezione del modello.

Poichè, come precedentemente descritto, la funzione score proposta vuole tenere in considerazione la complessità del JT corrispondente al modello selezionato, la selezione degli archi della rete è basata su una misura locale che risulta essere un compromesso tra la verosimiglianza dell'arco e la complessità del JT associato al sottografo che contiene l'arco. Una volta individuato l'insieme degli archi che soddisfano al processo di selezione del modello, le rete Bayesiana corrispondente è utilizzata per l'inferenza probabilistica: per fare ciò si considera il JT associato alla rete globale selezionata.

Capitolo 4

Risultati sperimentali

Si presentano i risultati ottenuti utilizzando la procedura euristica JT-Based proposta come metodo per la selezione del modello.

Per l'esperimento si è effettuata la simulazione di database di varie dimensioni da reti, utilizzate frequentemente in letteratura come riferimento per la ricerca riguardante le reti Bayesiane:

1. ASIA (Lauritzen e Spiegelhalter, 1988). Corrisponde alla versione semplicata di una rete utilizzata per la diagnosi di pazienti che arrivano in una clinica, dove ogni nodo corrisponde ad alcune condizioni del paziente. La rete complessiva è determinata da 8 nodi e 8 archi. Ogni variabile presenta due possibili stati (Figura 4.1).

2. ALARM (Beinlich et al., 1989). La rete è stata costruita come prototipo per la modellizazione delle cause in problemi legati all'anestesia. Essa è determinata da un totale di 37 nodi e 46 archi che rappresentano 8 problemi diagnostici, 16 cause e 13 variabili intermedie. Ogni nodo presenta un numero di possibili stati compreso tra 2 e 4. La costruzione della rete è avvenuta utilizzando la conoscenza soggettiva di un esperto sulle relazioni tra le variabili del dominio (Figura 4.2) .

Per ognuna delle precedenti reti Bayesiane sono stati generati sette database di diverse dimensioni. In particolare, per verificare il comportamento in relazione alla dimensione campionaria, sono stati simulati database con 100, 500, 1000, 5000, 10000, 15000 e 20000 casi[1]. Per fare ciò è stato utilizzato il software *Netica*, in grado di generare una serie di records casuali la cui distribuzione di probabilità si configura come quella della rete originaria da cui si simula (http://www.norsys.com).

[1]Per la rete ALARM sono stati considerati i database con numerosità campionaria fino a 10000 in quanto il pc utilizzato nell'analisi non riusciva, per mancanza di memoria, ad analizzare i rimanenti database.

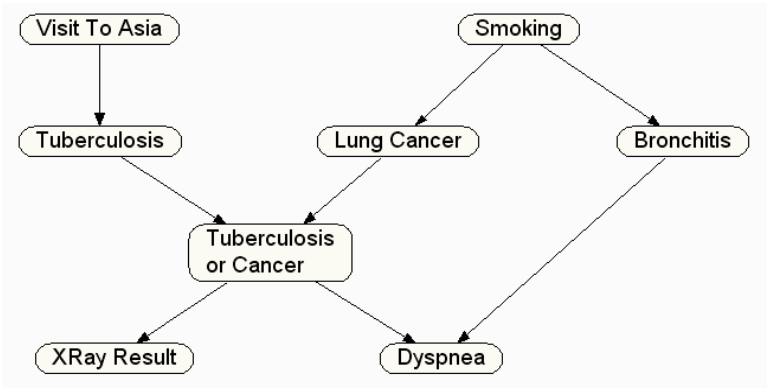


Figura 4.1. Rete Bayesiana: ASIA

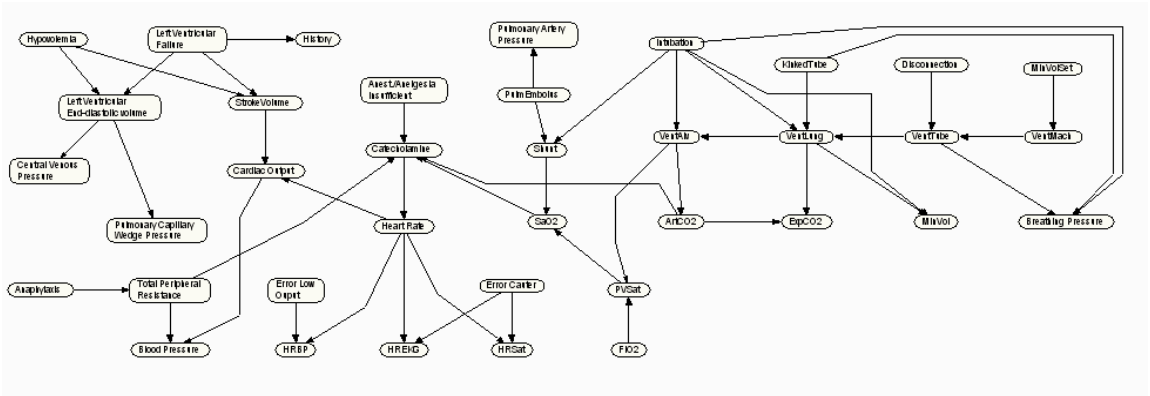


Figura 4.2. Rete Bayesiana: ALARM

Si confrontano i risultati del metodo proposto con quelli ottenuti utilizzando le funzioni score BIC, BDe e K2, con ordinamento delle variabili fissato[2] o non fissato[3]. Le procedure di ricerca utilizzate sono del tipo greedy Hill-Climbing (procedura K2 nel caso di ordinamento fissato e procedura K2SN nel caso di ordinamento non fissato).

Le statistiche usate per la valutazione sono relative ad aspetti significativi nell'analisi del comportamento dei metodi analizzati. In particolare, per valutare l'accuratezza della rete appresa si considera:

- Numero di archi extra A individuati in funzione della numerosità campionaria;
- Numero di archi mancanti D in funzione della numerosità campionaria;
- Numero di archi invertiti I in funzione della numerosità campionaria[4];
- Distanza di Hamming in funzione della numerosità campionaria. Questa misura, definita da $H = A + D + I$, riassume le tre misure precedentemente analizzate e fornisce una indicazione sulla capacità dell'algoritmo di apprendere una rete che risulti simile a quella da cui sono simulati i dati.

Per valutare la bontà del modello appreso rispetto ai dati, si considerano:

- Log-verosimiglianza della rete;
- Misura $KL(G,D)$ definita da:

$$KL(G,D) = \sum_{i=1,Pa(X_i)\neq\emptyset}^{n} MI(X_i,Pa(X_i))$$

  dove $MI(\mathbf{X},\mathbf{Y})$ è la misura della mutua informazione tra due insiemi di variabili $\mathbf{X}$ e $\mathbf{Y}$. La misura $KL(G,D)$ è una trasformazione monotona decrescente della distanza di Kullback tra la distribuzione di probabilità associata al database e la distribuzione di probabilità associata alla rete G (de Campos, 1998; Lam e Bacchus, 1994) in modo che più alto è il valore di $KL(G,D)$, migliore risulta l'adattamento della rete ai dati. Questa misura può indicare anche un overfitting del modello (una rete con molti archi ha valore $KL(G,D)$ alto).

Per lo scopo per cui è stato proposto l'utilizzo del metodo JT-Based, si considerano le caratteristiche associate alla dimensione del JT corrispondente alla rete appresa:

[2] Si considera l'ordinamento delle variabili che si deduce dalla rete originale.
[3] Si confrontano i risultati ottenuti con il metodo K2 solo nel caso di ordinamento fissato.
[4] Si considera questa statistica solo nel caso di ordinamento non fissato delle variabili.

- Confronto, per ogni dimensione campionaria, della percentuale di archi correttamente individuati e della percentuale della dimensione del JT corrispondente alla rete individuata dall'algoritmo, in rapporto alla dimensione del JT corrispondente alla rete originale.

Si considerano infine il numero di statistiche calcolate e il tempo di esecuzione:

- TEst: il numero totale di statistiche del tipo N_{ijk} valutate durante il processo di apprendimento.Questo valore non è necessariamente uguale al numero di statistiche realmente calcolate dai dati[5], poichè si possono usare tecniche di tipo *hashing* per evitare la necessità di ricalcolare valori precedentemente ottenuti, riducendo sostanzialmente questo numero.

- EstEv: numero di *differenti* statistiche usate. Questo numero può essere molto più piccolo del precedente.

- NVar: numero medio di variabili presenti nelle statistiche valutate. Si considera questo valore poichè il tempo richiesto per calcolare una statistica cresce esponenzialmente con il numero di variabili coinvolte.

- T(sec): il tempo, espresso in secondi, di esecuzione totale del'algoritmo. Questo valore risulta essere una misura relativa dell'efficienza, in quanto esistono molti fattori in grado di influenzare il tempo di esecuzione di un algoritmo, legati alle caratteristiche del pc su cui avviene l'esperimento.

Il processore utilizzato nell'esperimento è un Pentium 4 (CPU 3.20 GHz, 0.99 GB di RAM).

4.1 ASIA: analisi dei risultati

Si considera inizialmente il caso di ordinamento fissato delle variabili.

In Figura 4.3 sono riportati i risultati dell'accuratezza della struttura appresa.

L'utilizzo del metodo JT-Based porta alla determinazione di una rete Bayesiana con la caratteristica di non avere archi che non appartengono alla rete originale (numero di archi extra pari a zero). Lo stesso comportamento si ha utilizzando la funzione score BIC, mentre le funzioni score BDe e K2 tendono ad introdurre archi extra soprattutto a numerosità del database basse.

[5]Usualmente questo risulta essere il processo più *costoso* negli algoritmi di apprendimento di tipo Search & Score.

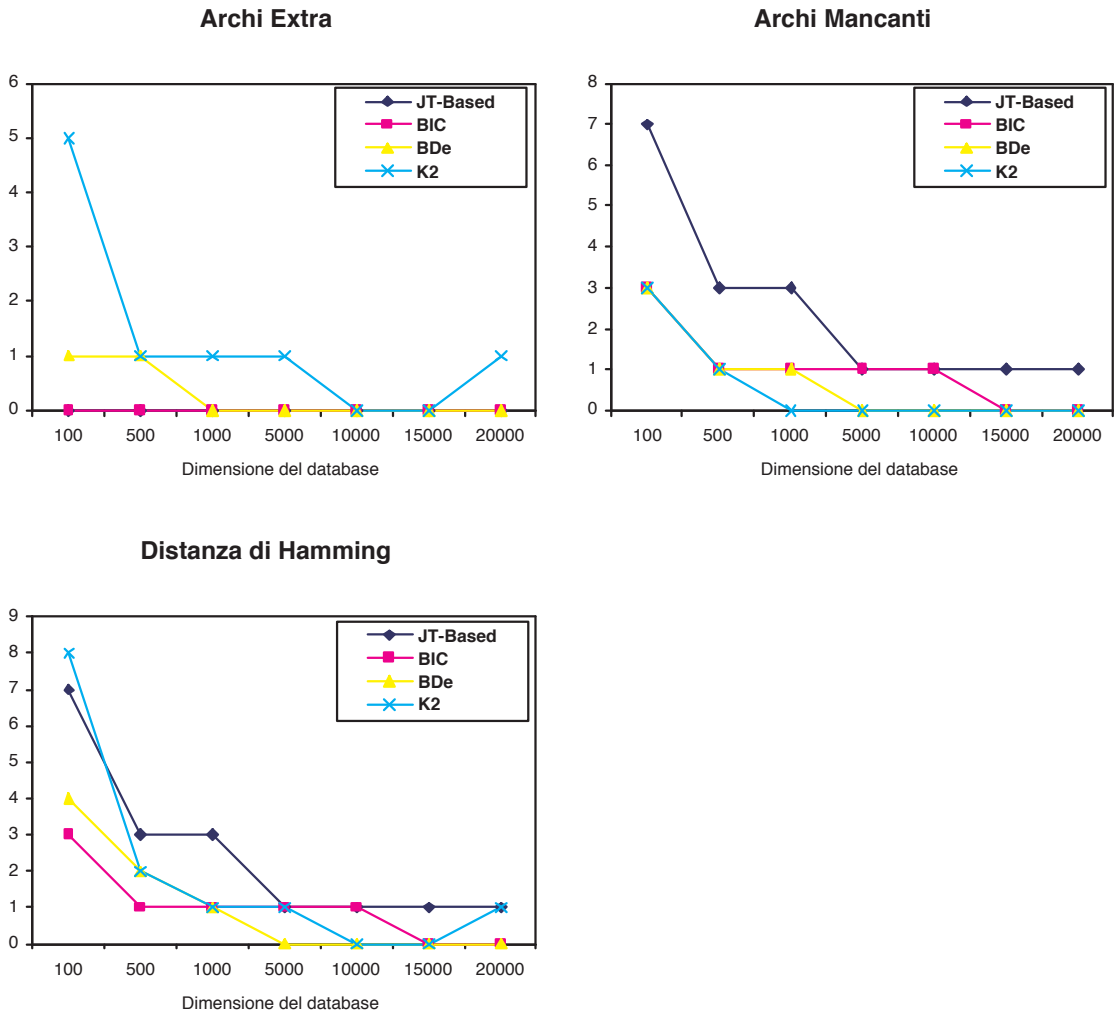


Figura 4.3. ASIA: accuratezza della struttura, ordinamento delle variabili fissato

		JT-Based	BIC	BDe	K2
Asia	100	-227,4442	-200,7035	-198,2475	-194,4719
	500	-1147,2243	-1119,2963	-1161,6607	-1114,334
	1000	-2204,7631	-2162,3184	-2162,3184	-2156,4235
	5000	-11110,3418	-11110,3418	-11107,2778	-11103,8852
	10000	-22303,0504	-22303,0504	-22299,3499	-22299,3499
	15000	-33416,6817	-33410,0262	-33410,0262	-33410,0262
	20000	-44956,6245	-44948,4184	-44948,4184	-44948,4184

Tabella 4.1. ASIA: Log-verosimiglianza della rete, ordinamento delle variabili fissato

Il metodo proposto presenta la caratteristica di avere un alto numero di archi mancanti a numerosità campionarie basse; questo valore diminuisce quando la dimensione del database aumenta. Ciò è spiegato dal fatto che più grande è il database, più informazione è disponibile e migliore la rete appresa. Il metodo JT-Based non è in grado di individuare un arco della rete anche a numerosità alte. Per l'arco in questione (VisitAsia→Tubercolosis) si è calcolato il valore della statistica χ^2 di dipendenza tra le variabili interessate, ottenendo un valore pari a 0.009: ciò indica una sostanziale indipendenza, che determina l'incapacità del metodo JT-Based di individuazione dell'arco.

L'analisi riassuntiva dell'accuratezza strutturale attraverso la distanza H mostra la capacità di adeguamento del metodo JT-Based, rispetto agli altri metodi, quando la numerosità campionaria cresce.

In Tabella 4.1 è riportato il valore della Log-verosimiglianza delle reti apprese.

Per database con basse numerosità campionarie, il valore della Log-verosimiglianza associato alle reti apprese con il metodo JT-Based risulta essere inferiore rispetto agli altri valori; ciò è direttamente collegato al fatto che il metodo presenta un numero di archi mancanti maggiore degli altri. Già a partire da N=5000, il valore della Log-verosimiglianza è uguale o si avvicina molto ai valori corrispondenti alle reti apprese con agli altri metodi e le differenze che si riscontrano sono minime (sono verosimilmente dovute alla mancanza dell'arco VisitAsia→Tubercolosis).

Ciò è confermato anche dall'analisi del comportamento della misura $KL(G,D)$ mostrato in Figura 4.4. Si vede infatti che da N=5000 le curve che rappresentano questa misura sono praticamente coincidenti.

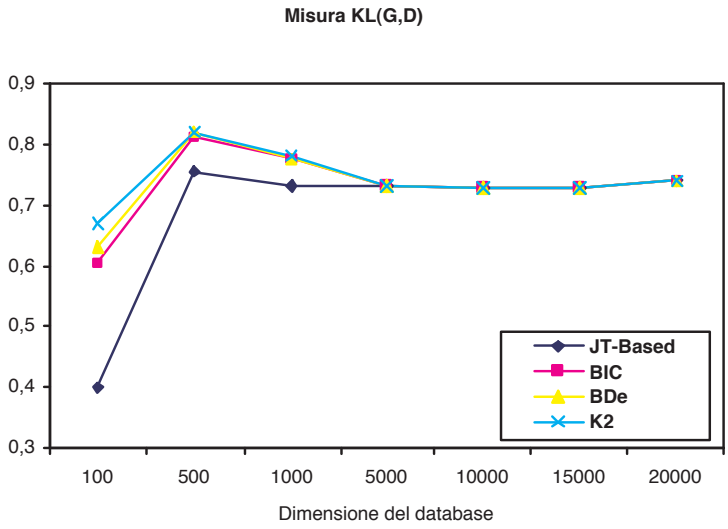


Figura 4.4. ASIA: Misura $KL(G,D)$, ordinamento delle variabili fissato

In Figura 4.5 viene mostrato, per ogni numerosità campionaria[6], la percentuale di archi correttamente individuata e il rapporto (in percentuale) della complessità del JT corrispondente alla rete appresa rispetto a quella della rete originale.

Per la rete ASIA, il JT è formato da 6 clique e ha una complessità pari a 40.

Per numerosità basse (N $\leq$1000) è verosimile che la percentuale della complessità del JT corrispondente alla rete appresa con il metodo JT-based sia inferiore rispetto alle altre a causa della bassa percentuale di archi correttamente individuati. Quando la numerosità campionaria aumenta e il metodo è in grado di individuare un alta percentuale di archi corretti, esso determina archi che mantengono bassa la complessità del JT.

Per esempio, per N=15000, si nota che la non inclusione dell'arco VisitAsia→Tubercolosis (come precedentemente affermato, l'analisi dei dati a disposizione dimostra che le due variabili risultano essere indipendenti) è in grado di ridurre la complessità del JT da 40 (complessità del JT corrispondente alla rete originale) a 38 (complessità del JT corrispondente alla rete appresa utilizzando il metodo JT-Based).

Si evidenzia come, anche se il numero di archi correttamente individuati è uguale, basta l'aggiunta di un solo arco per incrementare molto la complessità del JT: nel caso in cui N=5000, il metodo che utilizza la funzione score BIC e la funzione score K2 individuano tutti

[6]Non vengono riportati i risultati per N=20000 in quanto uguali al caso N=15000.

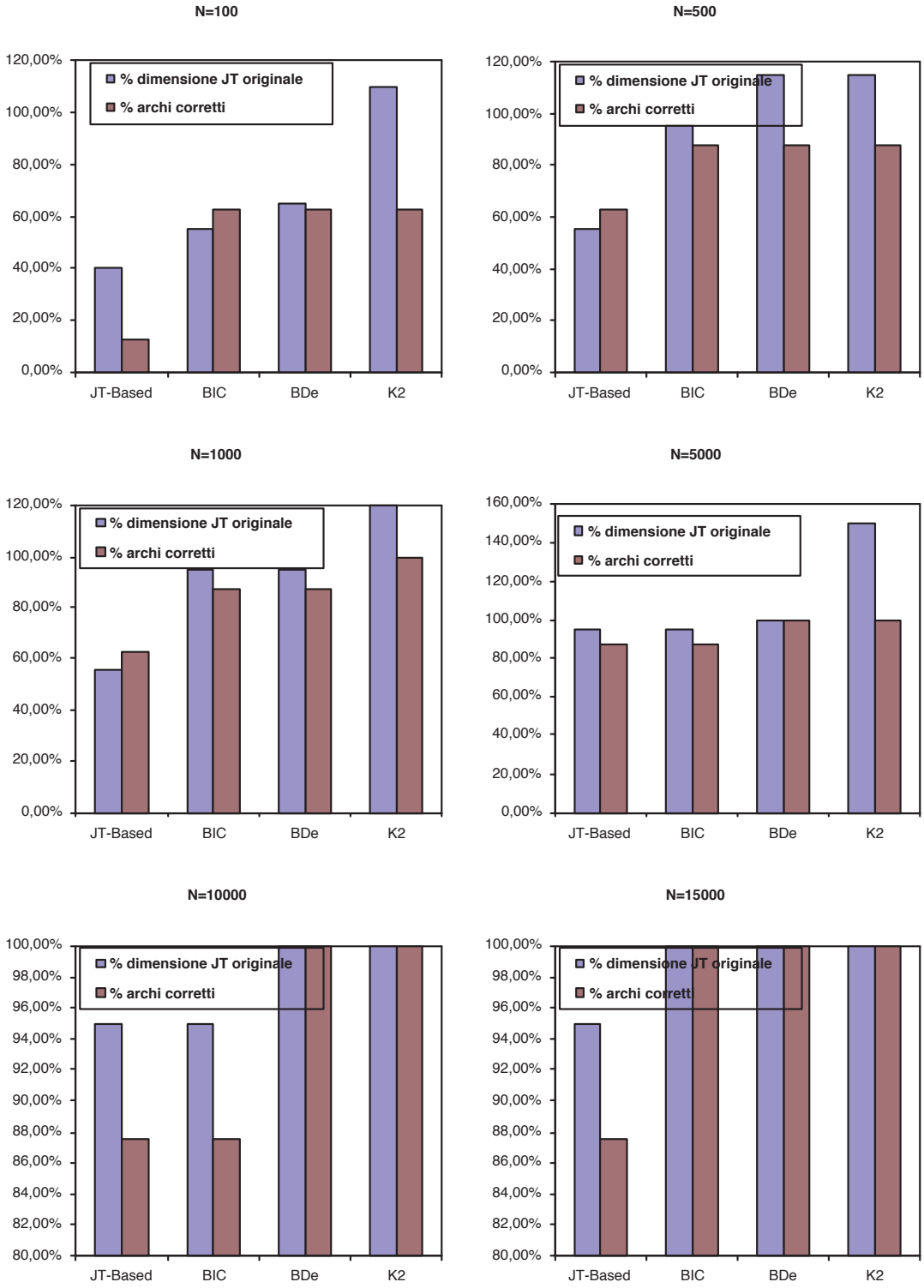


Figura 4.5. ASIA: Confronto tra la percentuale di archi correttamente individuata nelle rete la percentuale della dimensione del JT corrispondente rispetto a quello della rete originale, ordinamento delle variabili fissato.

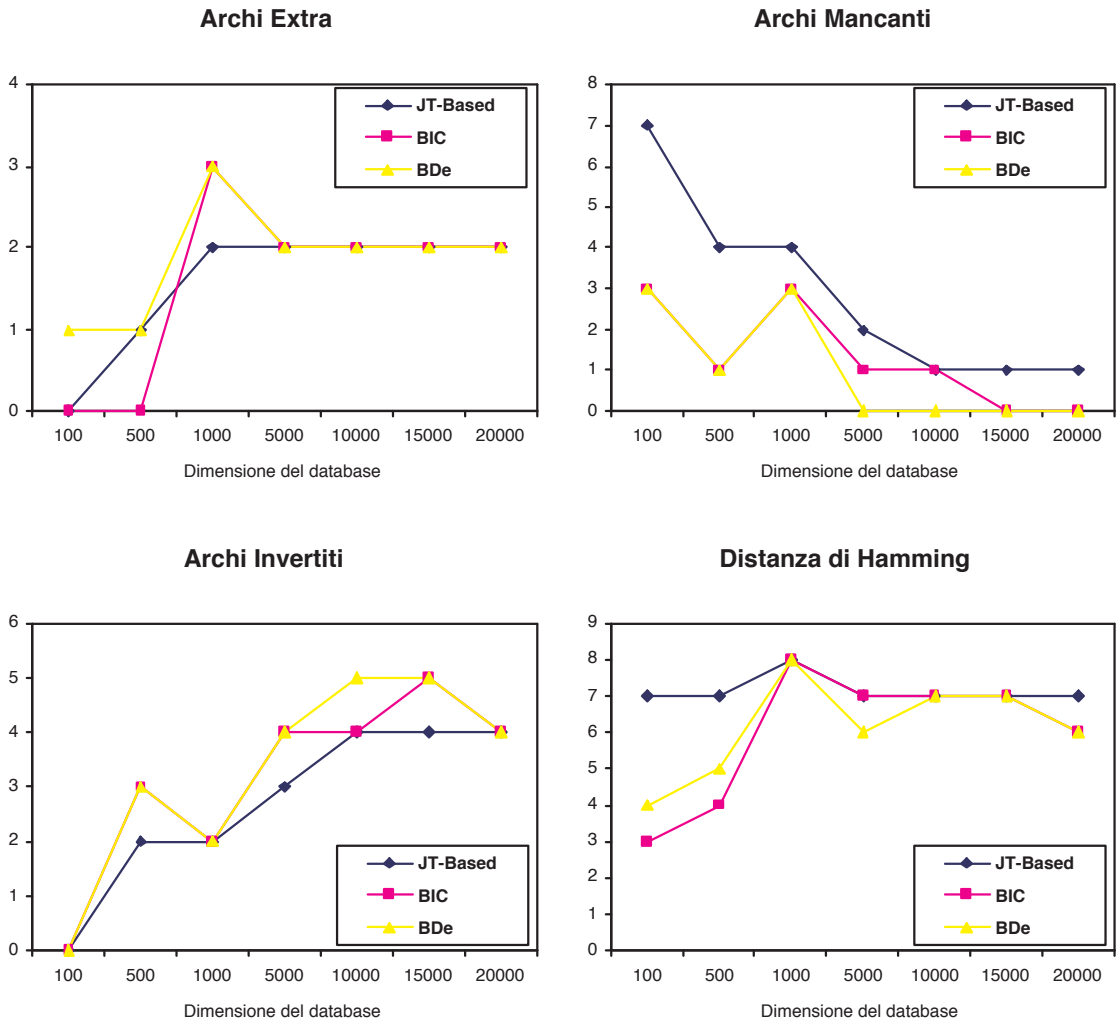


Figura 4.6. ASIA: accuratezza della struttura, ordinamento delle variabili non fissato

		JT-Based	BIC	BDe
Asia	100	-227,4442	-200,7035	-198,2475
	500	-1172,5104	-1123,8822	-1119,1519
	1000	-2207,0904	-2158,8225	-2161,5596
	5000	-11130,0484	-11108,5907	-11105,5267
	10000	-22306,1354	-22306,1354	-22302,4350
	15000	-33418,7187	-33412,0631	-33412,0631
	20000	-44958,5102	-44950,3041	-44950,3041

Tabella 4.2. ASIA: Log-verosimiglianza della rete, ordinamento delle variabili non fissato

gli archi della rete originale e, a questi, il K2 ne aggiunge uno. Quest'utimo ha associato un JT con complessità più grande rispetto all'originale in rapporto pari al 150%.

Usare un ordinamento porta a risultati migliori, in quanto l'ordinamento codifica informazione aggiuntiva. In Figura 4.6 si mostrano i risultati ottenuti togliendo il vincolo dell'ordinamento delle variabili.

Si nota come le reti apprese abbiano la tendenza ad avere un numero maggiore di archi extra rispetto al caso di ordinamento delle variabili. Il metodo JT-Based sembra comunque avere la proprietà di mantenere contenuto il numero di archi extra aggiunti nella rete. Riguardo al numero di archi mancanti, il comportamento del metodo proposto è simile al caso di ordinamento fissato delle variabili: il metodo presenta un numero elevato di archi mancanti a numerosità del database basse e non è in grado di individuare l'arco VisitAsia→Tubercolosis.

Rispetto ai risultati precedenti, si prende in considerazione anche il numero di archi invertiti: il metodo JT-Based risulta essere quello che fra tutti inverte il minor numero di archi. Considerando la distanza di Hammig, si nota come questo valore sia, per tutti i metodi, più alto rispetto al caso di variabili ordinate (si va a sommare il termine I), e che il metodo JT-Based presenta un comportamento simile agli altri anche a numerosità basse.

In Tabella 4.2 e in Figura 4.7 si confermano le valutazioni della bontà del modello appreso fatte nel caso dell'ordinamento fissato.

Sia la Log-verosimiglianza che la misura $KL(G,D)$ risultano essere simili; quando cresce la numerosità campionaria presentano valore inferiori per il fatto che il metodo JT-Based è in grado di determinare un numero minori di archi rispetto agli altri metodi.

Anche i risultati sulla complessità del JT delle reti apprese rispetto a quella originale rispecchiano le valutazioni fatte nel caso di ordinamento fissato come evidenziato nella Figura

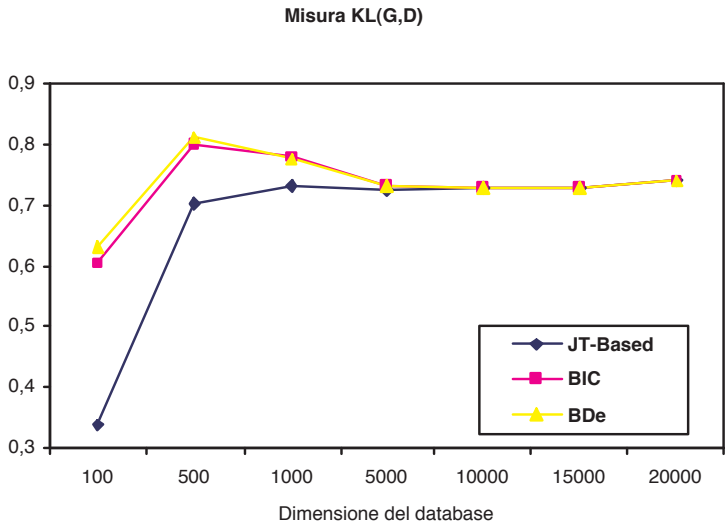


Figura 4.7. ASIA: Misura $KL(G,D)$, ordinamento delle variabili non fissato

4.8.

Nelle Tabelle 4.3, 4.4, 4.5 e 4.6, si riportano il numero di statistiche calcolate e il tempo di esecuzione per ogni metodo considerato, in modo da poter confrontare fra loro i metodi utilizzati e allo stesso tempo confrontare, per lo stesso metodo, la differenza che si ottiene fissando o non fissando un ordinamento per le variabili.

Nel confronto tra metodi in caso di ordinamento fissato delle variabili, non sono presenti differenze sostanziali sia per quanto riguarda il numero di statistiche calcolate, sia per il numero medio di variabili coinvolto in questi calcoli. Ciò è determinato dal fatto che i dati contengono molta informazione sulle caratteristiche del dominio, le variabili sono in numero limitato e tutti i metodi sono in grado di apprendere rapidamente una buona rete, come evidenziano i risultati descritti precedentemente. In particolare il metodo JT-Based è in grado di superare, in termini di tempo di esecuzione, sia il metodo BIC che il K2, dimostrando una sostanziale efficienza.

Nel caso senza ordinamento, il metodo JT-Based presenta caratteristiche che indicano una complessità minore rispetto agli altri, soprattutto se confrontata con il metodo BDe; il tempo di esecuzione dell'algoritmo è maggiore rispetto agli altri metodi nel caso di database con numerosità basse, mentre diminuisce fino a risultare più veloce, con numerosità alte. Questo è un buon risultato, in quanto se i dati da analizzare sono pochi, il tempo di esecuzione sarà comunque breve e dell'ordine di decimi di secondo, mentre se il database è grande (come

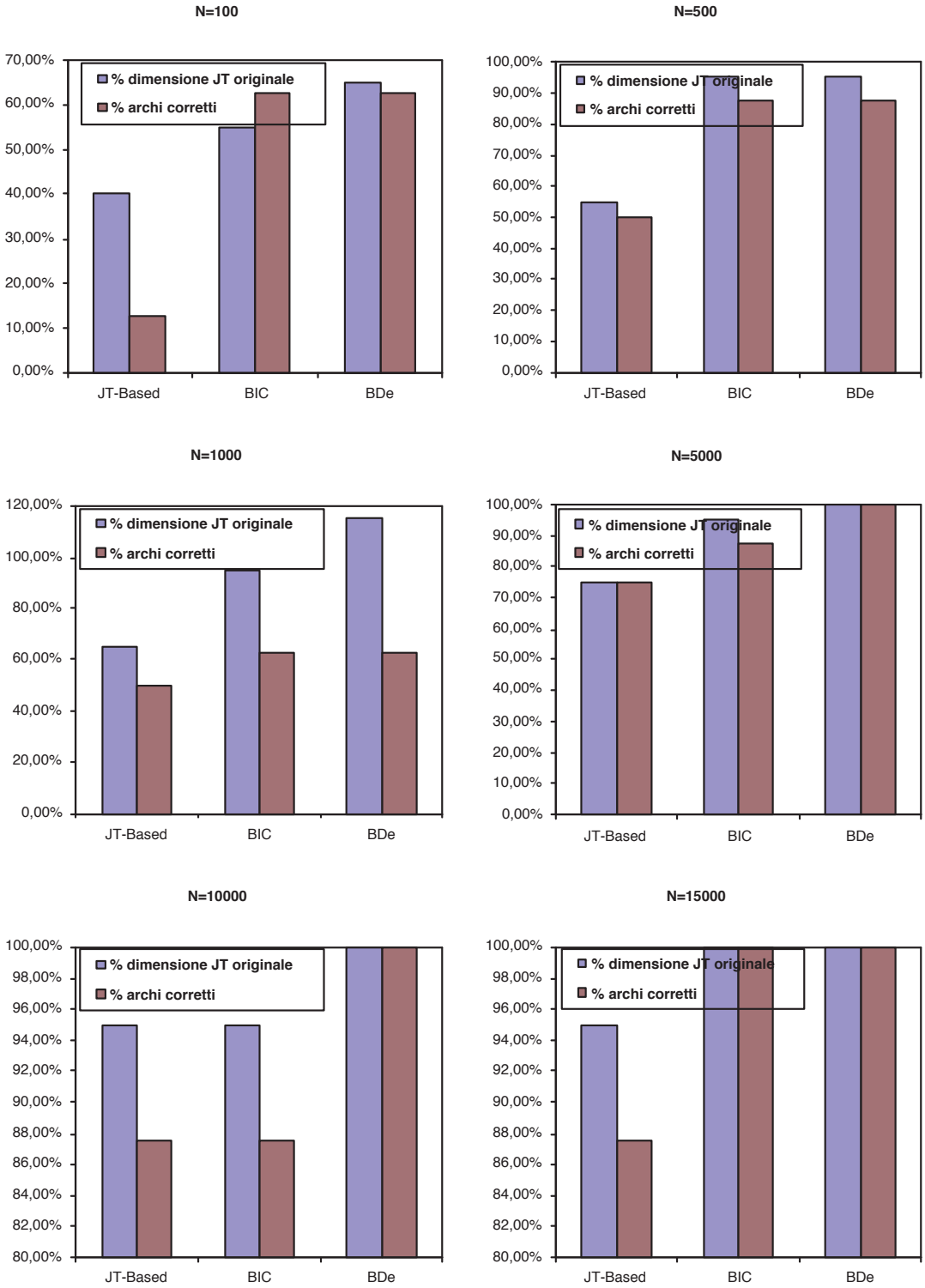


Figura 4.8. ASIA: Confronto tra la percentuale di archi correttamente individuata nelle rete la percentuale della dimensione del JT corrispondente rispetto a quello della rete originale, ordinamento delle variabili non fissato.

		con ordinamento				senza ordinamento			
		TEst	EstEv	NVar	T(sec)	TEst	EstEv	NVar	T(sec)
Asia	100	65	42	1,95	0,44	171	42	1,95	0,55
	500	83	56	2,25	0,97	193	55	2,22	0,67
	1000	83	56	2,25	0,70	203	58	2,26	0,79
	5000	92	63	2,41	1,27	241	84	2,65	1,83
	10000	92	63	2,41	1,97	241	84	2,65	2,84
	15000	92	63	2,41	2,56	247	88	2,74	3,79
	20000	92	63	2,41	3,16	243	85	2,66	4,59

Tabella 4.3. ASIA: Statistiche per il metodo JT-Based

		con ordinamento				senza ordinamento			
		TEst	EstEv	NVar	T(sec)	TEst	EstEv	NVar	T(sec)
Asia	100	83	56	2,25	0,03	193	57	2,25	0,03
	500	92	63	2,41	0,08	238	82	2,60	0,11
	1000	92	63	2,41	0,15	253	88	2,67	0,24
	5000	92	63	2,41	0,67	255	89	2,67	1,05
	10000	92	63	2,41	1,29	241	84	2,65	2,09
	15000	92	63	2,41	2,56	255	91	2,68	3,08
	20000	92	63	2,41	2,58	255	91	2,68	4,09

Tabella 4.4. ASIA: Statistiche per il metodo BIC

		con ordinamento				senza ordinamento			
		TEst	EstEv	NVar	T(sec)	TEst	EstEv	NVar	T(sec)
Asia	100	89	61	2,39	0,06	228	72	2,54	0,08
	500	97	67	2,51	0,28	245	83	2,71	0,36
	1000	92	63	2,41	0,47	253	88	2,67	0,70
	5000	92	63	2,41	2,23	255	89	2,67	3,42
	10000	92	63	2,41	4,41	255	91	2,68	6,91
	15000	92	63	2,41	2,56	255	91	2,68	10,28
	20000	92	63	2,41	8,78	259	94	2,75	14,53

Tabella 4.5. ASIA: Statistiche per il metodo BDe

		TEst	EstEv	NVar	T(sec)
Asia	100	100	69	2,61	0,08
	500	97	67	2,51	0,25
	1000	99	68	2,57	0,53
	5000	97	67	2,51	2,45
	10000	92	63	2,41	4,39
	15000	92	63	2,41	6,53
	20000	94	64	2,42	8,86

Tabella 4.6. ASIA: Statistiche per il metodo K2

nel caso di N=15000 o N=20000) ci si aspetta che il tempo di esecuzione aumenti, cosa che avviene utilizzando il metodo BDe: il metodo proposto riesce a contenere invece questo aumento.

Confrontando, infine, il comportamento di ogni singolo metodo con o senza ordinamento delle variabili, si vede come sia il metodo BIC che DBe presentano tempi di esecuzione sostanzialmente raddoppiati; il metodo JT-Based presenta invece un aumento minore rispetto agli altri metodi.

Per quanto riguarda il numero di statistiche calcolate, tutti i metodi hanno praticamente lo stesso tipo di incremento.

4.2 ALARM: analisi dei risultati

Considerare un dominio con poche variabili porta a risultati che non sono in grado di evidenziare chiaramente il miglioramento ottenuto utilizzando il metodo proposto; infatti, per quanto si sia riscontata una diminuzione sulla complessità del JT corrispondente alla rete appresa, non si può considerare elevata la complessità del JT originale.

Si considerano allora i risultati ottenuti per la rete ALARM, che presenta un numero elevato di variabili, di archi tra le variabili e una complessita del JT pari a 1065.

Nel caso di ordinamento fissato delle variabili, i risultati sull'accuratezza strutturale presentati in Figura 4.9 mostrano che ancora una volta il metodo JT-Based ha la capacità di introdurre il minor numero di archi extra, rispetto a tutti gli altri metodi, il che implica che fra tutti è quello che introduce minor informazione superflua nella rete. Anche in questo caso, presenta la caratteristica di avere un numero alto di archi mancanti per dimensioni del database basse.

Quando la numerosità cresce, il metodo JT-Based migliora il suo comportamento e produce risultati in linea con gli altri metodi.

Analizzando la distanza di Hamming si vede come, in generale da un punto di vista di ricostruzione strutturale della rete, il JT-Based non si allontana sostanzialmente dai risultati ottenuti dagli altri metodi e per dimensioni del database N$\geq$5000 presenta valore più basso tra tutti.

La valutazione della bontà di adattamento del modello ai dati attraverso l'analisi della Log-verosimiglianza in Tabella 4.7 mostra come nel caso N=100 il metodo JT-Based presenta un valore basso della Log-Verosimiglianza: ciò è determinato dal fatto che presenta un numero alto di archi mancanti. Nel caso N=500, si ha un adattamento ai dati migliore del metodo BIC, ma ancora sensibilimente diverso rispetto agli altri due. Quando la numerosità del database

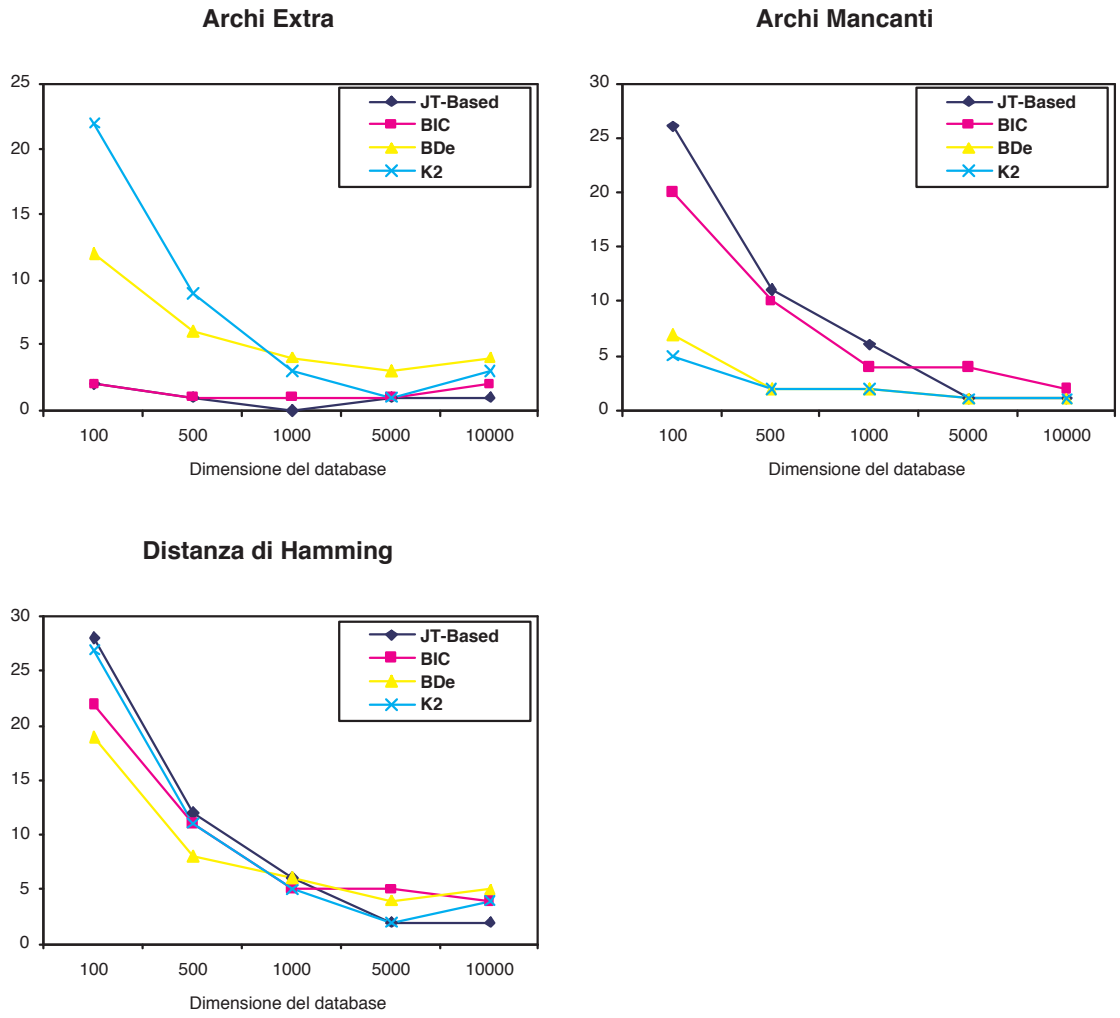


Figura 4.9. ALARM: Accuratezza della struttura, ordinamento delle variabili fissato.

		JT-Based	BIC	BDe	K2
Alarm	100	-1363,5736	-1244,5854	-1020,3628	-980,0274
	500	-4710,49287	-4780,7192	-4495,8955	-4493,7997
	1000	-9300,2104	-9415,4225	-9225,8986	-9228,299
	5000	-46195,4621	-46721,7634	-46190,7502	-46195,4621
	10000	-92811,2915	-93199,9883	-92798,8122	-92798,7364

Tabella 4.7. ALARM: Log-verosimiglianza della rete, ordinamento delle variabili fissato

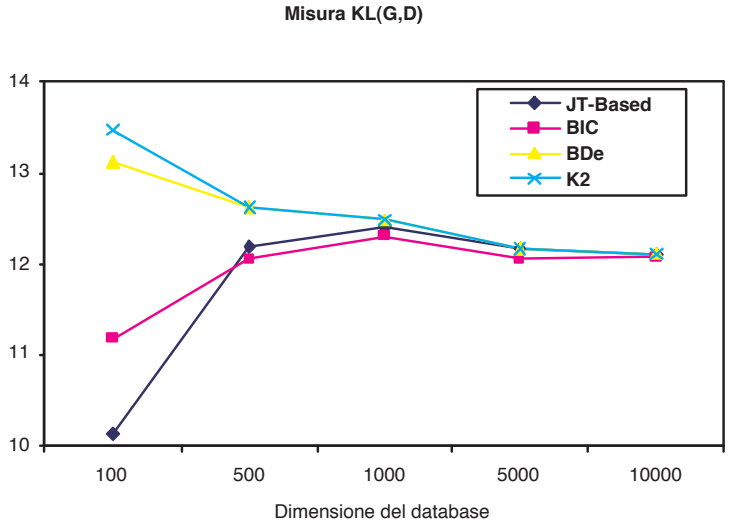


Figura 4.10. ALARM: Misura $KL(G,D)$, ordinamento delle variabili fissato.

aumenta, la Log-verosimiglianza assume valori buoni, in linea con i risultati ottenuti dagli altri metodi, continuando a superare il metodo BIC.

Questo andamento si riflette anche nella misura $KL(G,D)$ in Figura 4.10: già da N=500 il valore assunto dal metodo JT-Based supera il metodo BIC e procede, in linea con gli altri metodi, in modo tale che, a numerosità alte, le curve sono sostanzialmente coincidenti.

Nell'analisi della complessità del JT associato alle reti apprese, Figura 4.11, si evidenzia come anche piccoli cambiamenti nella struttura possono portare a grandi diversità nella complessità del JT. Per N=100, la bassa complessità del JT associato alla rete ottenuta con il metodo JT-Based dipende ancora una volta dal fatto che la percentuale di archi correttamente individuati è bassa. La stessa cosa si può dire per il metodo BIC.

Quello che sorprende, invece, è ciò che si verifica per il metodo BDe e per K2: avendo quest'ultimi determinato una alta percentuale di archi presenti nella rete originale, la differenza tra le complessità dei JT corrispondenti è molto alta, quasi quattro volte maggiore per il DBe e quasi nove volte per K2, a conferma del fatto che reti che presentano complessità strutturale simile possono avere complessità inferenziale molto diversa (Beygelzimer e Rish, 2002). Per N=500 questo aspetto si riproduce, anche se in maniera ridotta, mentre si nota che il metodo BIC presenta una complessità del JT corrispondente inferiore al JT-Based: ciò dipende dal fatto che basta anche solo l'aggiunta di un arco piuttosto che un altro per modificare in modo sensibile il JT e la corrispondente complessità. Il risultato si rovescia nel caso N=1000, in cui la diminuzione del numero di archi individuati rispetto al metodo BIC porta ad avere una complessità del JT minore. Per N=5000, si alza la percentuale di archi correttamente individuati (la rete presenta un arco extra e un arco mancante rispetto all'originale) e questo porta ad un aumento sostanziale della complessità del JT, pari ad una volta e mezzo quella del JT originale.

Il fattore positivo è che, a differenza dei metodo DBe, la caratteristica di includere nella rete un basso numero di archi extra[7], previene il fatto che il JT abbia complessità alta (quasi due volte e mezzo rispetto quella originale). La stessa cosa avviene per N=10000. Si nota inoltre che per N=5000 e N=10000, il metodo JT-Based e il K2 producono gli stessi risultati.

Il caso più interessante risulta essere quello in cui non si fissa un ordinamento per le variabili: si analizza un dominio in cui sono presenti molte variabili, come nel caso della rete ALARM, e non si dispone di nessun'altra informazione che quella contenuta nel database.

L'accuratezza strutturale, Figura 4.12, evidenzia come nel caso in cui le variabili non presentano un ordinamento fissato, il metodo JT-Based è quello che introduce il minor numero di archi extra nella rete e che presenta il maggior numero di archi mancanti.

Si può dunque affermare che basare la selezione del modello su una misura che considera

[7]Il metodo JT-Based include nella rete un arco extra, mentre il metodo BDe ne include tre.

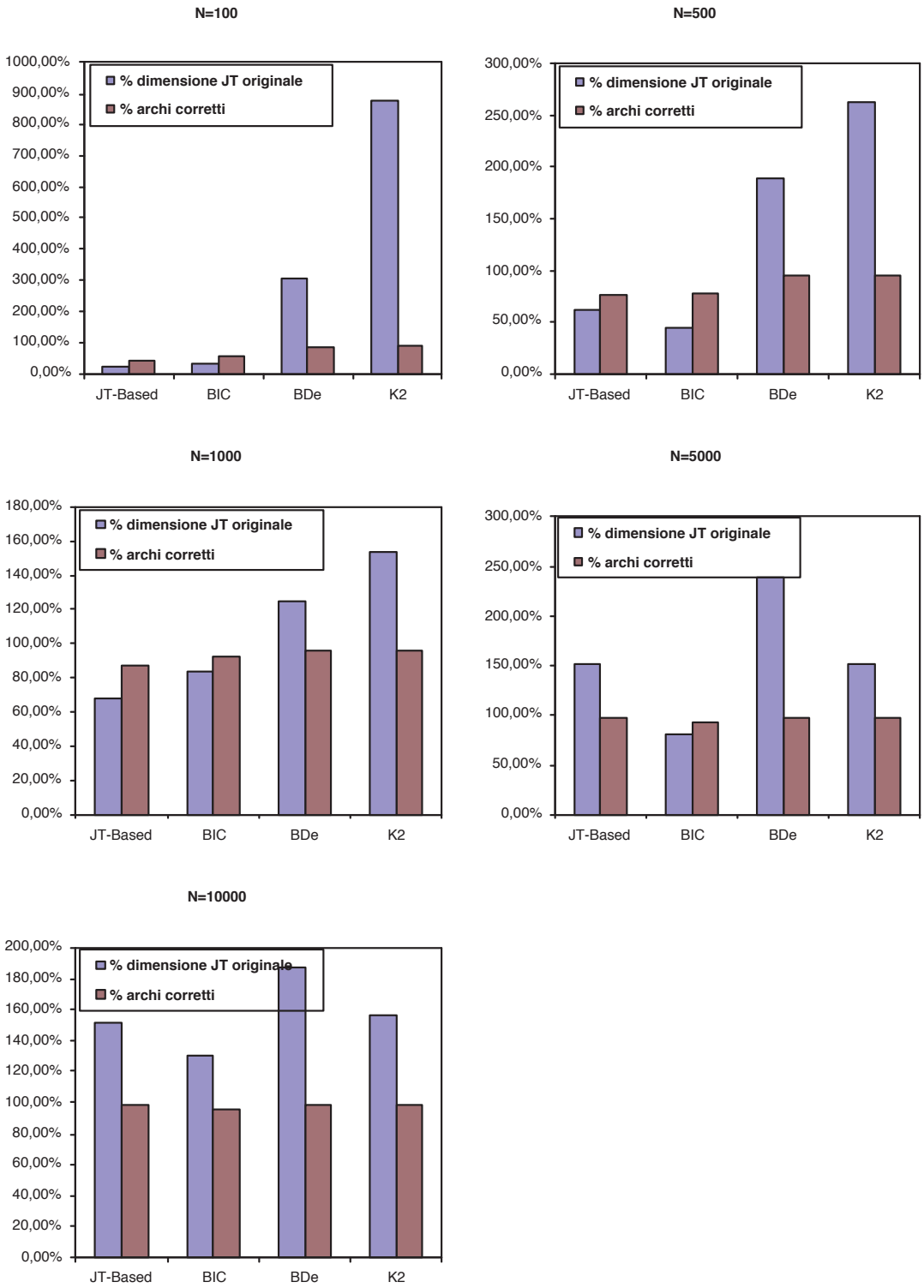


Figura 4.11. ALARM: Confronto tra la percentuale di archi correttamente individuata nelle rete la percentuale della dimensione del JT corrispondente rispetto a quello della rete originale, ordinamento delle variabili fissato.

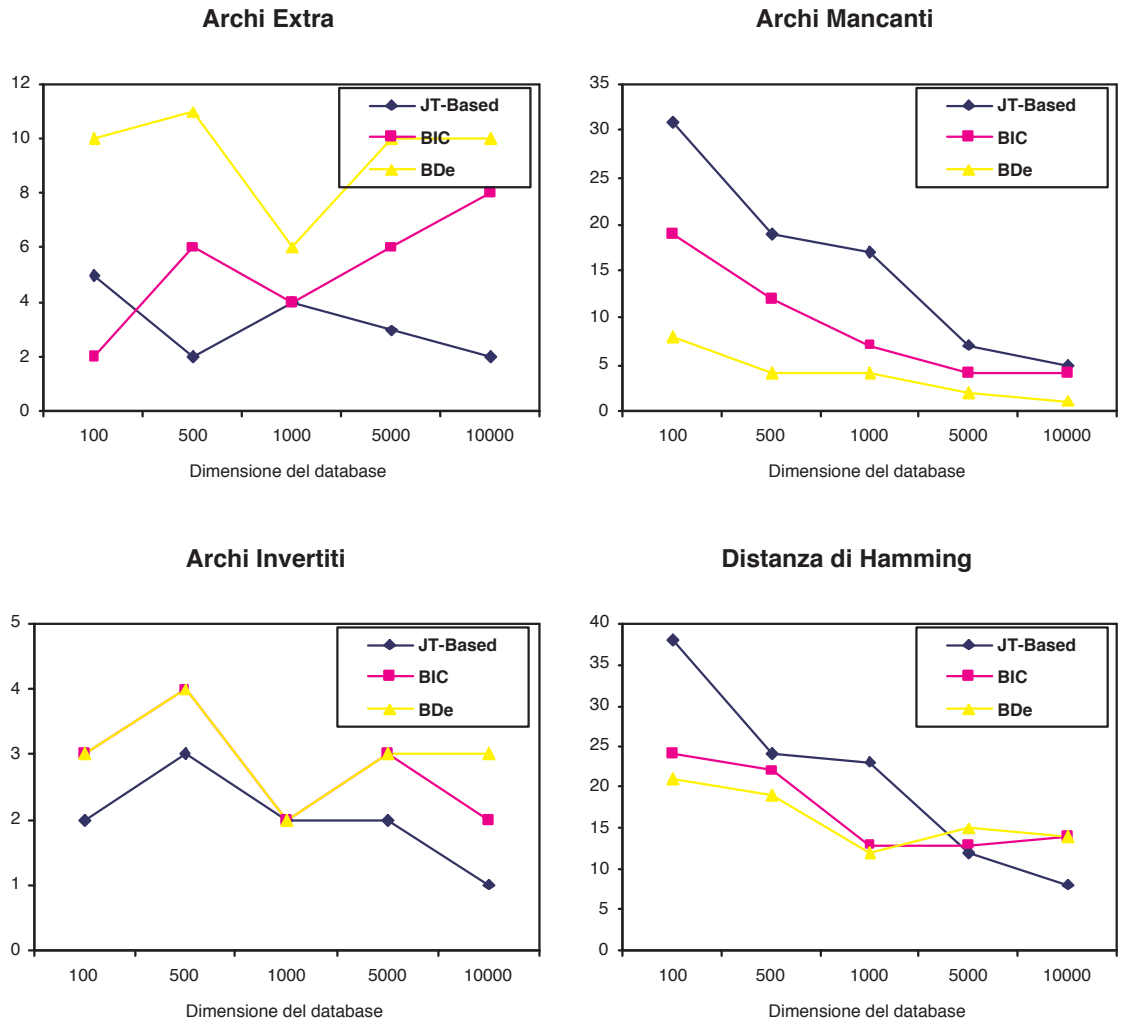


Figura 4.12. ALARM: Risultati dell'accuratezza, ordinamento delle variabili non fissato.

		JT-Based	BIC	BDe
Alarm	100	-1546,7008	-1240,8019	-1058,8729
	500	-5477,7000	-4851,8559	-4539,9427
	1000	-10604,0937	-9574,0513	-9368,0816
	5000	-47800,4772	-46927,3941	-46403,9235
	10000	-94822,9441	-93824,1114	-93237,8128

Tabella 4.8. ALARM: Log-verosimiglianza della rete, ordinamento delle variabili non fissato

la complessità del JT, porta all'apprendimento di reti più povere, in termini di numero di archi, rispetto a considerare altre funzioni score.

E' dunque vero che due reti che presentano la stessa complessità strutturale possono avere complessità del JT molto diversa, ma è vero anche che una bassa complessità del JT porta ad avere reti con bassa complessità strutturale. Si ha che:

$$\text{Complessità del JT bassa} \Rightarrow \text{Complessità strutturale della rete bassa}$$

ma non è vero il contrario

$$\text{Complessità strutturale della rete bassa} \not\Rightarrow \text{Complessità del JT bassa}$$

Un'altra caratteristica significativa del metodo proposto è che, come evidenziato anche per la rete ASIA, fra tutti, presenta il minor numero di archi invertiti.

La distanza di Hamming assume un valore inizialmente molto alto rispetto agli altri metodi, ciò a causa dell'alta numerosità di archi mancanti; per N=5000 e N=10000 il metodo JT-Based presenta invece il valore più basso fra i metodi analizzati, a dimostrazione del fatto che, complessivamente, è in grado di ricostruire la struttura originale in modo efficiente, non determinando tutti gli archi effettivi, ma limitando il numero di archi extra e invertiti.

Il basso numero di archi presenti nella rete appresa con il JT-Based porta ad avere valori della Log-verosimiglianza bassi, come si può notare in Tabella 4.8.

Anche l'analisi della misura $KL(G,D)$, Figura 4.13, evidenzia le stesse considerazioni.

Per motivare questo risultato, si consideri il numero di archi corretti, rispetto alla rete originale, individuati dagli algoritmi sul totale degli archi individuati, Figura 4.14. Una caratterizzazione della Log-verosimiglianza e della misura $KL(G,D)$ è quella di preferire reti con un numero alto di legami. Il metodo JT-Based tende invece ad includere pochi archi e di conseguenza il valore di queste misure risulta essere inferiore agli altri due metodi, in particolare al metodo BDe.

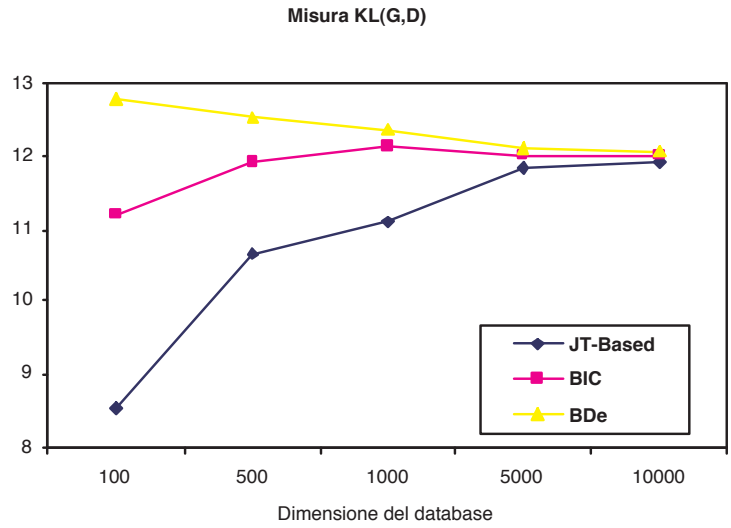


Figura 4.13. ALARM: Misura $KL(G,D)$, ordinamento delle variabili non fissato.

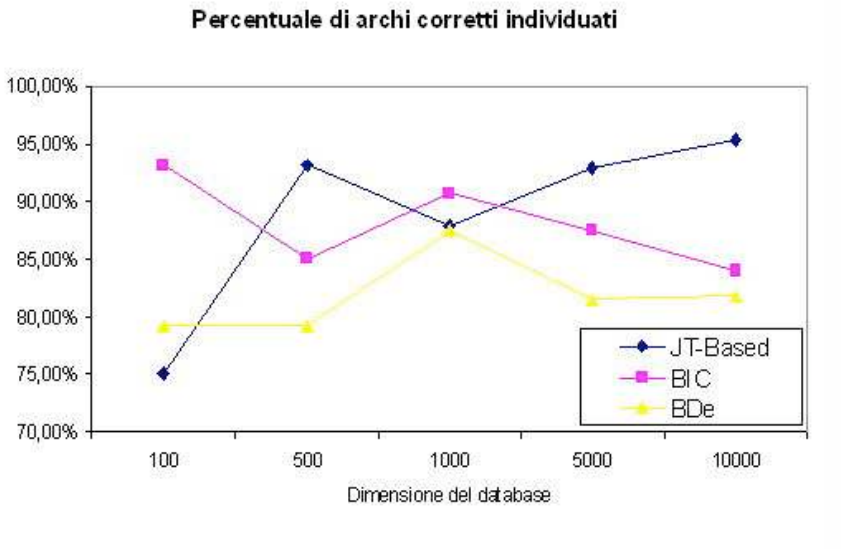


Figura 4.14. ALARM: Percentuale di archi correttamente individuati sul totale, ordinamento delle variabili non fissato

Analizzando la percentuale di archi corretti determinati dagli algoritmi sul totale di archi individuati, si evidenzia che il metodo JT-Based è quello che tra tutti inserisce meno archi, ma con una percentuale alta di correttezza: esso individua pochi archi, o comunque meno rispetto agli altri metodi, ma, di questi, quasi tutti sono esatti, a dimostrazione del fatto che il metodo è in grado di limitare l'aggiunta di informazione non esatta alla rete. Pur presentando dunque delle buone caratteristiche dal punto di vista della ricostruzione della struttura, ha valore di adattamento del grafo ai dati minore rispetto agli altri metodi considerati.

Nel confronto della dimensione della complessità del JT corrispondenti alle reti apprese con i diversi metodi, Figura 4.15, appare chiaro come il JT-Based sia in grado di determinare strutture a cui corrispondono JT con bassa complessità.

Nel caso N=100, N=500 e N=1000, questo può essere implicato dal fatto che la percentuale di archi correttamente individuata è minore rispetto agli altri metodi. Nel caso N=5000 e N=10000, la percentuale di archi correttamente individuati è pressochè uguale nei tre metodi confrontati, ma la complessità dei corrispondenti JT è molto diversa: nel caso del JT-Based, quest'ultima è approssimativamente pari a 2/3 della complessità del JT corrispondente alla rete originale, mentre questo valore è quasi una volta e mezza per il metodo BIC e più del doppio per il metodo DBe. Questi risultati risultano essere dunque significativi per lo scopo per il quale si è proposto l'utilizzo di questo metodo.

Il confronto dei valori del numero di statistiche calcolate e i tempi di esecuzione, riportate nelle Tabelle 4.9,4.10,4.11 e 4.12, dimostrano come, nel caso in cui sia fissato un ordinamento per le variabili, il comportamento dei metodi JT-Based e BIC sia sostanzialmente simile e superiore in termini di efficienza per BDe e il K2. Inoltre, a numerosità elevate, il numero di statistiche calcolate e il numero medio di variabili tendono ad essere simili in tutti i metodi. Ciò non si può dire per il tempo di esecuzione: sia a numerosità del database piccole che grandi, i secondi impiegati dal metodo BDe e da K2 sono fino a quattro volte più grandi di JT-Based e BIC.

Nel caso in cui non si fissi un ordinameneto tra i nodi, si evidenzia l'efficienza del metodo proposto: nel confronto con il metodo BIC, i valori delle statistiche presentate sono minori. La cosa è ancora più evidente nel confronto con il BDe. Anche il tempo di esecuzione dell'algoritmo si comporta allo stesso modo. Il numero di variabili coinvolte nell'analisi è sostanzialmente diverso nel caso in cui si consideri un ordinamento fissato o non fissato sulle variabili. Anche se in misura ridotta rispetto agli altri metodi, il tempo di esecuzione dell'algoritmo aumenta in alcuni casi più del doppio.

La scelta del metodo utilizzato per la selezione del modello dipende dall'uso che si intende fare della rete appresa. Quando una rete Bayesiana viene usata come punto di partenza per un'analisi esplorativa, può essere meglio considerare una rete che presenta una struttura con

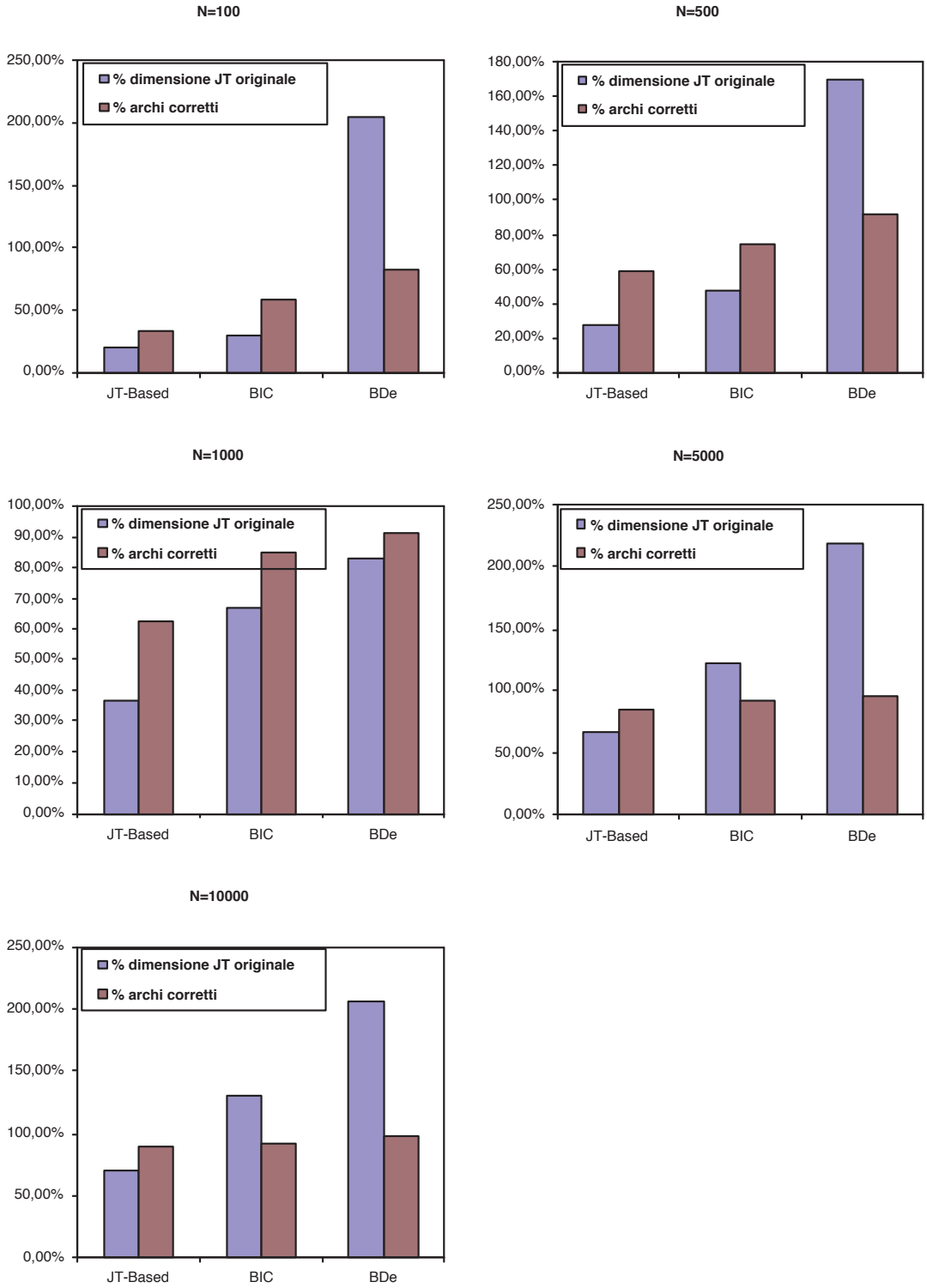


Figura 4.15. ALARM: Confronto tra la percentuale di archi correttamente individuata nelle rete la percentuale della dimensione del JT corrispondente rispetto a quello della rete originale, ordinamento delle variabili non fissato.

		con ordinamento				senza ordinamento			
		TEst	EstEv	NVar	T(sec)	TEst	EstEv	NVar	T(sec)
Alarm	100	1274	1143	2,37	1,33	13286	1384	2,47	1,64
	500	1547	1403	2,67	3,02	17379	2474	2,86	5,36
	1000	1641	1493	2,78	5,14	19322	2828	3,00	9,94
	5000	1758	1604	2,89	21,67	20586	3974	3,28	65,98
	10000	1758	1604	2,89	40,97	21394	4446	3,36	123,74

Tabella 4.9. ALARM: Statistiche per il metodo JT-Based

		con ordinamento				senza ordinamento			
		TEst	EstEv	NVar	T(sec)	TEst	EstEv	NVar	T(sec)
Alarm	100	1371	1235	2,48	0,62	17682	2492	2,83	1,59
	500	1548	1403	2,67	2,02	18677	3688	3,08	6,17
	1000	1667	1516	2,73	3,98	20328	4371	3,22	13,22
	5000	1673	1522	2,75	17,87	21850	4762	3,44	66,19
	10000	1757	1603	2,84	39.81	22674	5024	3,48	136,53

Tabella 4.10. ALARM: Statistiche per il metodo BIC

		con ordinamento				senza ordinamento			
		TEst	EstEv	NVar	T(sec)	TEst	EstEv	NVar	T(sec)
Alarm	100	1850	1691	3,02	3,51	22101	3745	3,34	7,08
	500	1817	1659	2,88	8,14	22178	5060	3,49	29,73
	1000	1779	1623	2,82	13,91	21947	5166	3,53	56,59
	5000	1791	1635	2,91	68,45	23415	54403	3,62	275,52
	10000	1826	1669	2,90	141,47	23685	5605	3,65	569,48

Tabella 4.11. ALARM: Statistiche per il metodo BDe

		TEst	EstEv	NVar	T(sec)
Alarm	100	2178	2007	3,25	4.84
	500	1905	1744	2,9621	8,86
	1000	1766	1611	2,82	14,95
	5000	1758	1604	2,89	68,27
	10000	1812	1656	2,91	137,23

Tabella 4.12. ALARM:Statistiche per il metodo K2

molte dipendenze fra i nodi, in modo da identificare successivamente quali, fra queste, sono meno significative o non sono supportate da evidenza empirica acquistata sul problema. In questo caso è bene utilizzare una funzione score Bayesiana (K2 o BDe) poichè l'utilizzo di questi metodi porta all'apprendimemento di reti dense, ovvero con un alto numero di archi tra le variabili.

Nel caso in cui la rete Bayesiana serva come modello per l'analisi delle (in)dipendenze condizionate che sussistono tra i dati, è importante considerare modelli che non siano densi, in quanto una lettura grafica delle relazioni risulterebbe improponibile. L'uso della score BIC, per come essa è definita, limita questo problema. Anche il nuovo metodo proposto, che considera la complessità del JT associato alla rete, previene questo problema.

Ma è nel caso in cui la rete Bayesiana serve come strumento di inferenza probabilistica che l'utilizzo del metodo JT-Based mostra la sua efficienza: esso infatti determina una rete il cui associato JT presenta una complessità bassa, in modo tale che la complessità del processo inferenziale, direttamente collegato a questa misura, sia anch'essa bassa.

4.3 Applicazione a database reali

Si presentano i risultati ottenuti applicando la procedura JT-Based proposta a database reali. Sono stati considerati i domini reali utilizzati nelle analisi descritte in Brogini et al. (2004) e in Bolzan et al. (2005).

In Brogini e al. (2004), si presenta una analisi esplorativa che intende ricercare e descrivere le variabili e gli indicatori associati ad esperienze familiari come il ricovero ospedaliero, con particolare attenzione al livello di soddisfazione percepito dal paziente ricoverato in ospedale.

Il database utilizzato è tratto dall'indagine multiscopo sulle famiglie fatta dall'ISTAT nel 1998 *Famiglie, soggetti sociali e condizioni d'infanzia*: attraverso la selezione da parte di

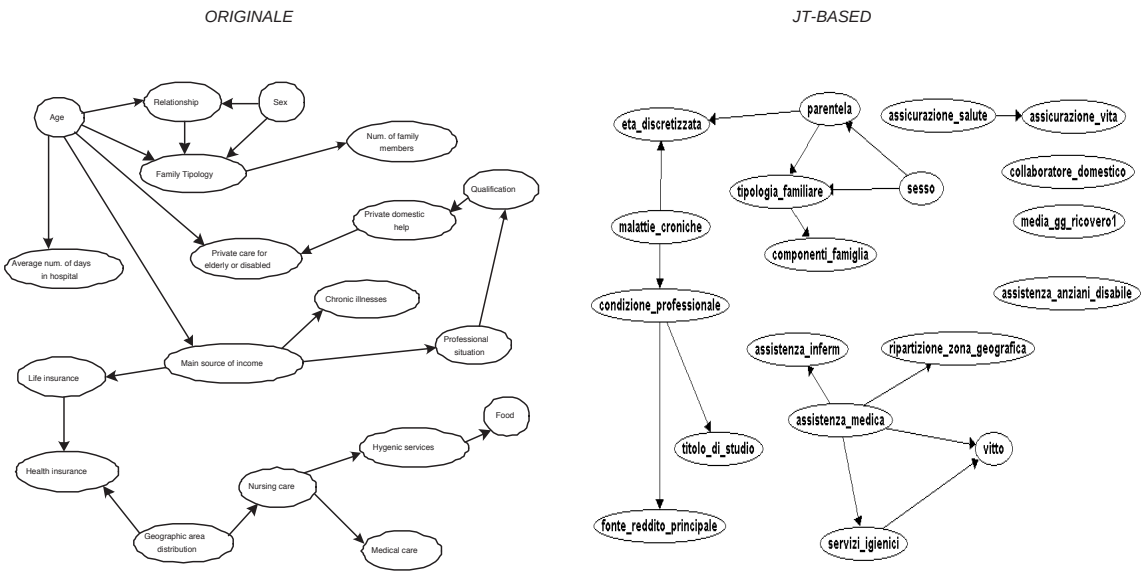


Figura 4.16. Reti Bayesiane reali

un esperto di 19 variabili discrete o opportunamente discretizzate, alcune delle quali sono trasformazione di quelle originali dell'indagine, si arriva alla costruzione di un database che non contiene dati mancanti e che presenta un totale di record pari a 2.197.

L'apprendimento della rete è avvenuto utilizzando il metodo K2 e la conoscenza (soggettiva) dell'esperto.

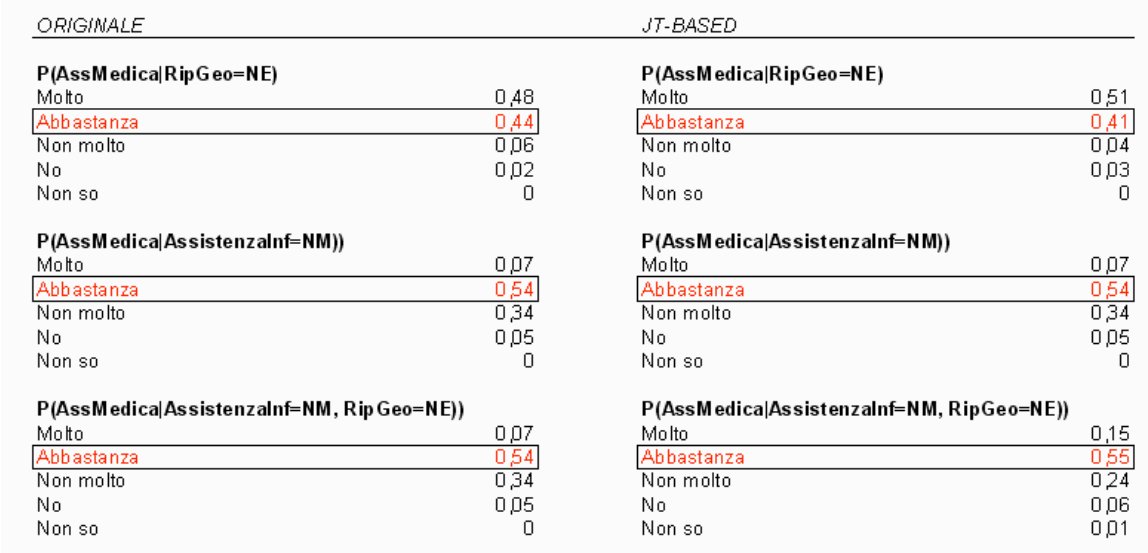
In Figura 4.16 si presentano le reti apprese: a sinistra è rappresentata la rete originale, mentre a destra è raffigurata la rete Bayesiana appresa utilizzando il metodo JT-Based[8].

	Archi	Log-Vs	KL(G,D)	BIC	BDe	K2	JT
ORIGINALE	21	-34455,76	3,95	-37261,36	-36239,31	15522,08	1206
JT-BASED	15	-35352,00	3,55	-37233,95	-36849,74	-15782,13	**546**

Tabella 4.13. Statistiche per le reti Bayesiane reali

In Tabella 4.13, sono riportate alcune statistiche significative nel confronto delle due reti apprese.

[8]I nodi nella rete originale hanno nomi in inglese, cosi come sono descritti nell'articolo citato di riferimento.



ORIGINALE		JT-BASED	
P(AssMedica|RipGeo=NE)		**P(AssMedica|RipGeo=NE)**	
Molto	0,48	Molto	0,51
Abbastanza	0,44	Abbastanza	0,41
Non molto	0,06	Non molto	0,04
No	0,02	No	0,03
Non so	0	Non so	0
P(AssMedica|AssistenzaInf=NM))		**P(AssMedica|AssistenzaInf=NM))**	
Molto	0,07	Molto	0,07
Abbastanza	0,54	Abbastanza	0,54
Non molto	0,34	Non molto	0,34
No	0,05	No	0,05
Non so	0	Non so	0
P(AssMedica|AssistenzaInf=NM, RipGeo=NE))		**P(AssMedica|AssistenzaInf=NM, RipGeo=NE))**	
Molto	0,07	Molto	0,15
Abbastanza	0,54	Abbastanza	0,55
Non molto	0,34	Non molto	0,24
No	0,05	No	0,06
Non so	0	Non so	0,01

Figura 4.17. Probabilità condizionate per le reti Bayesiane in esame

La rete appresa con il metodo JT-Based presenta un numero inferiore di archi e una Log-verosimiglianza in valore più bassa rispetto alla rete originale considerata nell'analisi. La differenza di bontà di adattamento del modello ai dati riscontrata risulta essere debole, come si verifica anche dall'analisi della misura $KL(G,D)$.

In Tabella 4.13 sono riportate anche il valore delle misure BIC, BDe, e K2 calcolate per le due reti apprese.

Nell'ultima colonna, infine, è riportata la complessità del JT: la complessità del JT corrispondente alla rete appresa con il nuovo metodo proposto risulta essere pari a metà della complessità del JT appreso nell'analisi originale. Ciò implica che la complessità del processo inferenziale, che utilizza la rete Bayesiana appresa con il metodo JT-Based, è inferiore e dunque più efficiente.

In Brogini et al. (2004) si deduce che, per l'obiettivo considerato, ovvero l'analisi della soddisfazione del paziente nei confronti del sistema di assistenza medica, i due fattori principali in grado di determinare il livello di soddisfazione risultano essere rappresentati dalle variabili che identificano la ripartizione geografica d'appartenenza del paziente e il livello di soddisfazione legato al sistema di assistenza infermieristica.

Per valutare come cambia l'inferenza probabilistica nelle due reti, si è calcolata la probabilità condizionata della variabile che codifica il livello di soddisfazione rispetto al sistema di assistenza medica con riferimento alla ripartizione geografica, al livello di soddisfazione

riguardo al sistema di assistenza infermieristica e congiuntamente a questi due aspetti. I risultati ottenuti sono mostrati in Figura 4.17.

Le probabilità calcolate risultano essere molto simili, in alcuni casi sono uguali, il che dimostra che la rete appresa con il nuovo metodo è in grado di condurre agli stessi risultati ma con una efficienza, relativa al processo di inferenza probabilistica, doppia rispetto alla rete considerata nell'analisi originale.

Inoltre si nota come nella rete originale, il condizionamento legato al livello dei soddisfazione riguardo al sistema di assistenza infermieristica sia fondamentale per il blocco dell'informazione verso la variabile oggetto di studio: nessuna altra informazione disponibile nel sistema influenza la distribuzione di probabilità della variabile studiata, una volta che si conosce lo stato assunto dalla variabile corrispondente al livello di soddisfazione riguardo l'assistenza infermieristica. Nella rete appresa con il metodo JT-Based, pur essendo ancora il condizionamento legato a quest'ultima variabile a determinare sostanzialmente la distribuzione di probabilità della variabile d'interesse, l'informazione derivante dalla conoscenza sullo stato delle altre variabili modifica, seppur in modo esiguo, la distribuzione di probabilità: questo effetto è un aspetto più che plausibile nell'analisi di un sistema complesso.

Il secondo database reale considerato (Bolzan et al. 2005) viene utilizzato per analizzare i fattori e le relazioni del sistema di assistenza informale necessario nel caso di ricovero di un paziente in ospedale. Anche in questo caso il database deriva dall'indagine multiscopo sulle famiglie fatta dall'ISTAT nel 1999-2000 *Sulle Condizioni di Salute e Ricorso ai Servizi Sanitari della Popolazione Italiana*.

Si sono selezionate 21 variabili discrete o opportunamente discretizzate, alcune della quali sono una riclassificazione di quelle originali dell'indagine, ottenendo un database che non contiene dati mancanti e con un totale di record pari a 5056.

La rete è appresa utilizzando l'algoritmo di apprendimento K2 e la conoscenza (soggettiva) di un esperto.

In Figura 4.18 si presentano le reti apprese: a sinistra è rappresentata la rete originale, mentre a destra è raffigurata la rete Bayesiana appresa utilizzando il metodo JT-Based.

In Tabella 4.14, sono riportate alcune statistiche significative nel confronto fra due reti.

La rete appresa con il metodo JT-Based, presenta anche in questo caso un numero inferiore di archi e una Log-verosimiglianza in valore più bassa rispetto alla rete originale considerata nell'analisi.

La misura $KL(G,D)$ ha valore praticamente uguale a conferma del fatto che le due reti presentano la stessa distanza di Kullback tra la distribuzione di probabilità associata al grafo e la distribuzione di probabilità associata al database. Si riportano anche i valori delle misure BIC, BDe, e K2 calcolate per le due reti apprese.

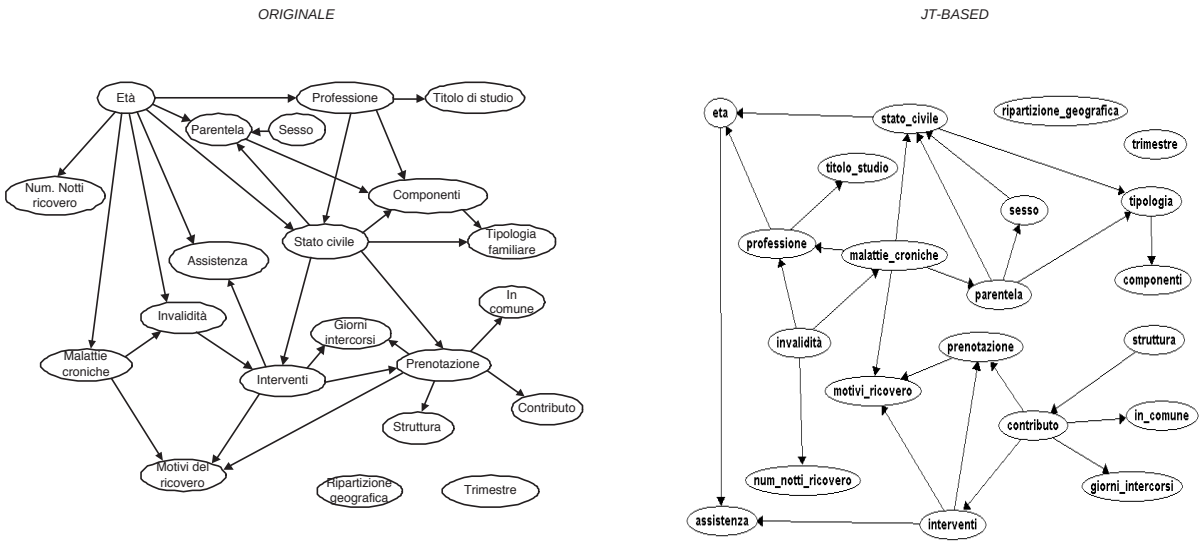


Figura 4.18. Reti Bayesiane reali

	Archi	Log-Vs	KL(G,D)	BIC	Bayes	K2	JT
ORIGINALE	30	-85348,68	4,32	-87135,37	-87034,48	-37614,18	840
JT-BASED	26	-87107,41	4,33	-88267,26	-88254,91	-38193,53	**492**

Tabella 4.14. Statistiche per le reti Bayesiane reali

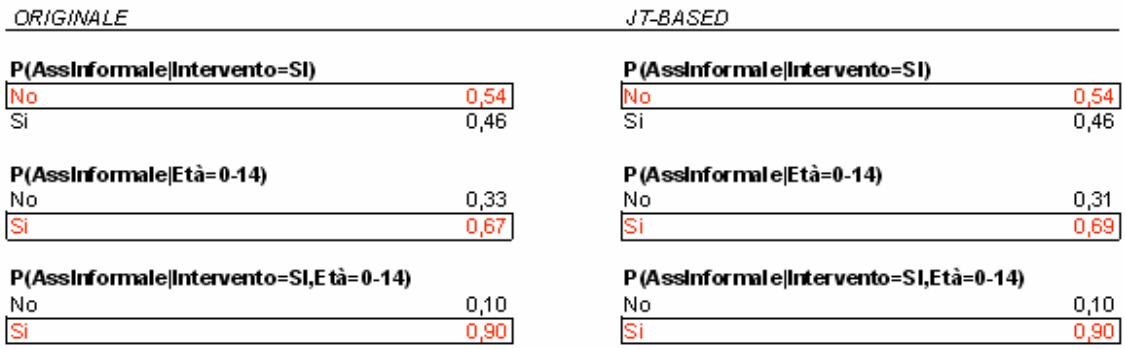


Figura 4.19. Probabilità condizionate per le reti Bayesiane in esame

La complessità del JT corrispondente alle reti, evidenziata nell'ultima colonna della Tabella 4.14, mostra come il metodo proposto sia in grado di apprendere reti con complessità inferiori, in questo caso pari quasi alla metà della complessità del JT corrispondente alla rete dell'analisi originale.

Nell'analisi condotta, uno degli obiettivi principali risultava essere, come precedentemente illustrato, l'individuazione dei fattori principali legati all'assistenza informale.

In Bolzan et al. (2005) si individuano l'età del paziente ricoverato e la presenza di intervento chirurgico come le variabili più caratterizzanti nella determinazione dell'assistenza informale. Per questo, si calcolano le distribuzioni di probabilità condizionate della variabile che rappresenta l'assistenza informale, rispetto alle variabili che descrivono l'età, la presenza di intervento chirurgico e ad entrambe queste variabili.

I risultati ottenuti, presentati in Figura 4.19, dalla rete considerata nell'analisi originale, sono uguali a quelli ottenuti dalla rete appresa con il nuovo metodo proposto. Poichè anche la distribuzione di probabilità associata al numero di notti di ricovero risultava essere un fattore significativo nell'analisi, si presentano i risultati relativi a questo aspetto condizionatamente ai due fattori più caratterizzanti: l'età del paziente ricoverato e la presenza di invalidità.

Come si nota dalla Figura 4.20, le distribuzioni di probabilità ottenute sono uguali e il condizionamento, relativo alla variabile età, blocca l'informazione proveniente da altre variabili. Ciò avviene in entrambe le reti a dimostrazione del fatto che si possono ottenere gli stessi risultati significativi anche considerando reti Bayesiane con strutture diverse: la complessità inferenziale bassa non influenza il comportamento della rete in termini di significatività di risultati.

ORIGINALE

P(NumNottiRicovero|Età=0-14)

0-4	0,64
5-7	0,25
8-14	0,06
15+	0,05

P(NumNottiRicovero|Invalidità=NO)

0-4	0,46
5-7	0,22
8-14	0,18
15+	0,13

P(NumNottiRicovero|Età=0-14, Invalidità=NO)

0-4	0,64
5-7	0,25
8-14	0,06
15+	0,05

JT-BASED

P(NumNottiRicovero|Età=0-14)

0-4	0,46
5-7	0,23
8-14	0,19
15+	0,13

P(NumNottiRicovero|Invalidità=NO)

0-4	0,43
5-7	0,22
8-14	0,20
15+	0,15

P(NumNottiRicovero|Età=0-14, Invalidità=NO)

0-4	0,46
5-7	0,23
8-14	0,19
15+	0,13

Figura 4.20. Probabilità condizionate per le reti Bayesiane in esame

Capitolo 5

Conclusioni e ricerche future

L'incertezza è una delle prerogative del ragionamento umano e per questo motivo, numerosi contesti della vita reale si configurano in termini di sistemi complessi multidimensionali, affetti da incertezza. La quantificazione dell'incertezza è resa possibile attraverso l'uso della Teoria della Probabilità e della conoscenza a disposizione, intesa come informazione utile per la sua specificazione.

Uno strumento attraverso cui si formalizzano questi aspetti è la rete Bayesiana: una combinazione di elementi di Teoria dei grafi, di Teoria della Probabilità e di Metodologia Statistica che permette la selezione di un modello per elaborare situazioni complesse, in cui sia presente incertezza.

Attraverso la semantica delle reti Bayesiane, si semplifica il processo di inferenza probabilitica, ovvero il calcolo delle probabilità di un evento che coinvolge variabili del dominio condizionatamente a qualsiasi altro evento. Le metodologie, usate per l'inferenza nelle reti Bayesiane, si differenziano tra loro per il tipo di struttura assunta dalla rete. In generale, la struttura di una rete è rappresentata da un grafo DAG, nel quale almeno due nodi sono connessi da più di un cammino, comportando una notevole complessità degli algoritmi per l'inferenza. Esistono molti metodi proposti in letteratura per il calcolo dell'inferenza, sia esatti che approssimati. La metodologia *Junction Tree Propagation*, è definita da un processo per il quale la propagazione dell'evidenza, ovvero la nuova informazione sullo stato delle variabili del dominio, può essere fatta in modo efficiente rappresentando la distribuzione di probabilità congiunta attraverso l'uso di un grafo indiretto detto *Junction Tree*, che si deriva dalla rete originale.

E' necessario dunque disporre della rete Bayesiana, che rappresenta il modello probabilistico sul dominio oggetto di studio. Per come è definita una rete Bayesiana, l'apprendimento si sviluppa attraverso metodi diversi a seconda che il grafo della rete sia noto o non noto:

nel primo caso si parla di apprendimento dei parametri, mentre nel secondo caso di apprendimento congiunto della struttura e dei paramentri della rete. Il database, fonte primaria dell'informazione, può presentare dati mancanti: in questo caso la letteratura propone approcci basati sulla modifica delle tecniche usate nel caso di dati completi. I principali approcci di apprendimento nel caso di struttura ignota e dati completi risultano: l'approccio *Search & Score* e l'approccio *Constraint-based*. Il primo ricerca la struttura che massimizza una determinata funzione score, definita in generale come misura di adattamento del modello ai dati; il secondo determina, attraverso l'uso di misure di indipendenza, usualmente test statistici, le relazioni di dipendenza e indipendenza tra le variabili del dominio e ricerca la rete che rappresenta al meglio queste relazioni. I metodi Constraint-based hanno lo svantaggio di essere molto sensibili ad errori nella conduzione delle verifiche di indipendenza: errori, anche in numero contenuto, possono portare alla determinazione di strutture sensibilmente diverse. Per questo, i metodi Search & Score sono più utilizzati nelle applicazioni, fornendo risultati più accurati. Idealmente si vorrebbe ricercare esaustivamente all'interno dello spazio di tutte le possibili strutture, obiettivo che è stato provato essere NP-hard. Si utilizzano algoritmi di ricerca euristici di tipo greedy, che analizzano lo spazio attraverso operatori locali quali l'aggiunta, la cancellazione o il cambio della direzione di un arco. La selezione del modello avviene usualmente in base a misure (funzioni score) calcolate direttamente sulla rete Bayesiana in esame.

L'approccio innovativo proposto e sviluppato in questa tesi, denominato *JT-Based*, basa l'apprendimento della struttura di una rete Bayesiana su una misura collegata alla complessità del processo inferenziale. Come precedentemente descritto, l'inferenza probabilistica è uno degli obiettivi che segue alla costruzione del modello. Più il modello è complesso, in termini sia di numerosità che di interpretazione dei legami della rete, più è naturale pensare che il processo coinvolto nel calcolo dell'inferenza sia complesso. Se si utilizza l'approccio basato sulla Junction Tree Propagation, la complessità dell'inferenza probabilistica è legata alla complessità di una struttura secondaria, il Junction Tree corrispondente alla rete. La nuova misura risulta essere un compromesso tra la bontà di adattamento del modello ai dati e la complessità del JT su cui verrà basata l'inferenza.

Per valutare il comportamento del metodo proposto, sono stati simulati database a partire dalla struttura nota di due reti Bayesiane, denominate ASIA e ALARM, frequentemente utilizzate in letteratura come strumento di analisi nelle ricerche relative alla teoria delle reti Bayesiane. Si sono generati, per ogni rete, sette database di dimensione diversa, in modo da testare la sensibilità del metodo alla numerosità campionaria. Si è valutato il comportamento della nuova metodologia attraverso il confronto con i risultati ottenuti utilizzando alcuni fra i metodi più frequentemente utilizzati in letteratura: gli algoritmi di tipo greedy Hill-Climbing

basati su funzioni score BIC e BDe, e l'algoritmo K2.

Si sono valutati i risultati nel caso in cui venga fissato un ordinamento sull'insieme delle variabili e nel caso in cui non si abbia alcun ordinamento prefissato. I confronti e le valutazioni sul comportamento del nuovo metodo sono stati effettuati considerando differenti aspetti significativi dell'analisi:

- Si è valutata la capacità di individuare la struttura da cui sono stati simulati i dati, la rete originale, attraverso indicatori specifici riferiti alla numerosità di archi mancanti, extra ed invertiti. Dai risultati ottenuti, si evince che l'accuratezza strutturale della rete appresa è in linea con quella degli altri metodi di raffronto utilizzati.

- Si è presa in considerazione la bontà di adattamento del modello ricavato dai dati simulati: a numerosità elevate, si ottengono valori non significativamente diversi.

- Si è valutata la complessità del processo inferenziale, misurata attraverso la complessità del JT corrispondente alla rete appresa. Si evidenzia come, nella maggior parte dei casi, i risultati sono in linea con le aspettative per cui è stato proposto l'approccio. Si ottengono infatti reti Bayesiane a cui corrispondono JT con complessità minore. Questo risultato produce un notevole vantaggio nel processo inferenziale con una riduzione sensibile di calcoli necessari per la specificazione della probabilità di un evento condizionatamente all'evidenza, rendendo il processo più efficiente.

- Si sono presentate le statistiche temporali della complessità del metodo, che confermano i risultati precedenti.

- L'approccio JT-Based è stato altresì valutato su reti Bayesiane ricavate da database reali, oggetto di analisi esplorative in studi applicativi condotti in precedenza, ottenendo una sostanziale conferma della sua validità.

L'uso dell'approccio JT-Based ovvia al problema che si ha nella definizione di una funzione score come compromesso tra l'adattamento del modello ai dati e la complessità del JT corrispondente alla rete appresa, che risulta non scomponibile in fattori locali. Il suo utilizzo comporterebbe il calcolo, ad ogni passo successivo, della complessità del JT corrispondente alla struttura generata. E' stata perciò definita una procedura euristica per la soluzione del problema, che porta ad una approssimazione di una funzione score del tipo descritto.

Sebbene i risultati ottenuti siano soddisfacenti, il campo di ricerca sull'argomento è aperto. Ricerche future si concentreranno:

- Sulla determinazione di quanto questa procedura locale sia in grado di fornire i migliori risultati. Per fare ciò, il processo prevede la costruzione e la successiva simulazione di

strutture *ad hoc* è per il problema, ovvero reti Bayesiane che presentano una misura della complessità strutturale bassa ma una misura della complessità dell'associato JT molto alta. Attraverso l'applicazione del metodo proposto ai database simulati, si valuterà quanto la procedura locale è in grado di produrre i risultati sperati (questo esperimento è gia in fase d'opera e il risultati ottenuti fino ad ora, confermano il buon comportamento del metodo JT-Based evidenziato nei risultati sperimentali descritti nella tesi).

- Sulla individuazione e definizione di una misura che consideri la complessità del JT globale, mantenendo l'efficienza dei calcoli inferenziali, pur non godendo della proprietà di scomponibilità. Sarà preferibile sperimentare l'approccio su reti Bayesiane con un numero relativamente basso di variabili, in modo da contenere temporalmente l'algoritmo di ricerca. Si può pensare di utilizzare successivamentele tecniche di *Incremental Learning* (Flores et al., 2003), in grado di determinare il JT corrispondente ad una rete Bayesiana, a cui vengono applicati piccoli cambi della struttura attraverso la determinazione della sotto-struttura del JT, che effettivamente ha subito una modifica.

Per quanto riguarda possibili ricerche legate alle applicazioni pratiche di reti Bayesiane in ambito dei sistemi complessi, l'interesse è rivolto a:

- L'utilizzo delle proprietà delle reti per la determinazione di variabili esplicative rilevanti nell'analisi di una variabile o più variabili di interesse (Brogini e Slanzi, 2005). Attraverso studi di simulazione, si dovrà caratterizzare quanto la proprietà di Markov Blanket sia in grado di determinare l'insieme minimo condizionante attraverso cui la/e variabile/i risulta indipendente da tutte le altre. L'insieme trovato potrà essere utilizzato, senza perdita di informazioni, ai fini di classificazione o di analisi statistiche successive.

- L'approfondimento dell'approccio Constraint-based, tipicamente per problemi di individuazione di relazioni causali, con l'utilizzo di test statistici parametrici e non parametrici.

- Lo studio del problema dei dati mancanti, sviluppando tecniche che permettano l'apprendimento, sia parametrico che strutturale, nel caso in cui l'ipotesi MAR non sia verificata.

Appendice A

Richiami di Teoria dei grafi

Si presentano gli elementi della Teoria dei grafi utilizzate nella tesi. Questa Teoria è ben definita in letteratura (Berge, 1958; Berge, 1973; Golumbic, 1980) e si sviluppa con lo scopo di fornire uno strumento astratto di manipolazione e di analisi, applicabile in differenti contesti.

Definizione A.1 ***Grafo***
Sia $\mathcal{V}$ un insieme finito non vuoto. Sia $\mathcal{A} \subset \mathcal{V} \times \mathcal{V}$. Un grafo G su $\mathcal{V}$ è definito attraverso la coppia

$$G=(\mathcal{V},\mathcal{A})$$

dove:

- *gli elementi di $\mathcal{V}$ sono detti nodi o vertici di G;*
- *gli elementi di $\mathcal{A}$, ovvero coppie ordinate di vertici, sono detti archi di G.*

Generalmente i nodi sono indicati con l'insieme $\mathcal{V}=\{v_1,v_2,\ldots,v_n\}$, e gli archi con la forma (v_i,v_j). La differenza fondamentale tra tipi di grafo dipende dalla natura degli elementi di $\mathcal{A}$. In particolare, si ha:

Definizione A.2 ***Arco diretto***
Sia $G=(\mathcal{V},\mathcal{A})$ un grafo. Quando $(v_i,v_j) \in \mathcal{A}$ e $(v_j,v_i) \notin \mathcal{A}$, il legame tra i nodi v_i e v_j è detto arco diretto. Un arco diretto tra due nodi si indica con $v_i \rightarrow v_j$.

Definizione A.3 ***Arco indiretto***
Sia $G=(\mathcal{V},\mathcal{A})$ un grafo. Quando $(v_i,v_j) \in \mathcal{A}$ e $(v_j,v_i) \in \mathcal{A}$, il legame tra i nodi v_i e v_j è detto arco indiretto. Un arco indiretto tra due nodi si indica con v_i —v_j o v_j —v_i.

Questa differenziazione tra tipi di elementi di $\mathcal{A}$, permette di definire i due principali tipi di grafo:

Definizione A.4 ***Grafo diretto e indiretto***
Un grafo $G=(\mathcal{V},\mathcal{A})$ è un grafo diretto se e solamente se tutti gli elementi di $\mathcal{A}$ sono archi diretti.
G è un grafo indiretto se e solamente se tutti gli elementi di $\mathcal{A}$ sono archi indiretti.

In un grafo diretto, perciò, l'ordine dei nodi che definisce un arco è importante, mentre in un grafo indiretto, questo ordine risulta essere irrilevante.

Definizione A.5 ***Cammino***
Sia $G=(\mathcal{V},\mathcal{A})$ un grafo. Una successione di nodi $[v_0,v_1,\ldots,v_n]$ è un cammino in G se e solamente se

$$\forall i,1 \leq i \leq n, \text{ si ha } (v_{i-1},v_i) \in \mathcal{A} \text{ oppure } (v_i,v_{i-1}) \in \mathcal{A}$$

Definizione A.6 ***Cammino semplice***
Sia $G=(\mathcal{V},\mathcal{A})$ un grafo. Un cammino $[v_0,v_1,\ldots,v_n]$ è un cammino semplice in G se e solamente se

$$\forall i,1 \leq i < j \leq n, \text{ si ha } v_i \neq v_j$$

Quindi per lo stesso nodo si passa al massimo una volta.

Definizione A.7 ***Cammino chiuso***
Un cammino è detto essere chiuso se il nodo di partenza coincide con quello di arrivo, ovvero $v_0 = v_n$.

Definizione A.8 ***Connessione***
Sia $G=(\mathcal{V},\mathcal{A})$ un grafo. Siano v_0 e v_n due nodi di G. Si dice che v_0 e v_n sono connessi se e solamente se esiste un cammino $[v_0,v_1,\ldots,v_n]$.

Definizione A.9 ***Grafo connesso***
Sia $G=(\mathcal{V},\mathcal{A})$ un grafo. G è un grafo connesso se, tra ogni coppia di nodi, esiste almeno un cammino.

Definizione A.10 ***Ciclo***
Sia $G=(\mathcal{V},\mathcal{A})$ un grafo diretto. Un ciclo è un cammino chiuso in G.

Definizione A.11 ***Grafo diretto aciclico (DAG)***
Un grafo diretto è detto aciclico se non contiene cicli.

Definizione A.12 ***Ciclo indiretto***
Sia G=($\boldsymbol{\mathcal{V}}$,$\boldsymbol{\mathcal{A}}$) un grafo indiretto. Un ciclo indiretto è un cammino chiuso in G.

Definizione A.13 ***Sottografo e grafo parziale***
Sia G=($\boldsymbol{\mathcal{V}}$,$\boldsymbol{\mathcal{A}}$) un grafo. Sia $W \subset \boldsymbol{\mathcal{V}}$ e $F \subset \boldsymbol{\mathcal{A}}$.
Si chiama sottografo di G il grafo $G_W = (W,\boldsymbol{\mathcal{A}} \cap W \times W)$.
Si chiama grafo parziale di G il grafo $G_F = (\boldsymbol{\mathcal{V}},F)$.

Definizione A.14 ***Catena***
Sia G=($\boldsymbol{\mathcal{V}}$,$\boldsymbol{\mathcal{A}}$) un grafo diretto. G è una catena se e solamente se ogni nodo di G ha al più un genitore e al più un figlio.

Definizione A.15 ***Albero***
Sia G=($\boldsymbol{\mathcal{V}}$,$\boldsymbol{\mathcal{A}}$) un grafo. G è un albero se e solo se è connesso e senza cicli.

Definizione A.16 ***Polialbero***
Sia G=($\boldsymbol{\mathcal{V}}$,$\boldsymbol{\mathcal{A}}$) un grafo. G è un polialbero se esiste al massimo una catena tra ogni coppia di nodi.

Si nota che un albero è un grafo non necessariamente diretto mentre un polialbero implica che G sia diretto. I polialberi godono delle seguenti proprietà:

Teorema A.1 *(Berge, 1973)*
Per qualsiasi grafo G=($\boldsymbol{\mathcal{V}}$,$\boldsymbol{\mathcal{A}}$), le proposizioni seguenti sono equivalenti:

1. *G è un albero.*
2. *G è un grafo connesso senza cicli.*
3. *G è connesso e $|\boldsymbol{\mathcal{A}}| = |\boldsymbol{\mathcal{V}}| - 1$.*
4. *G è connesso e minimale per $|\boldsymbol{\mathcal{A}}|$.*
5. *G è aciclico e $|\boldsymbol{\mathcal{A}}| = |\boldsymbol{\mathcal{V}}| - 1$.*
6. *G è aciclico e massimale per $|\boldsymbol{\mathcal{A}}|$.*
7. *$\forall v_i,v_j \in \boldsymbol{\mathcal{V}}$, esiste un'unica catena da v_i a v_j.*

8. *Tutti i grafi parziali di G sono non connessi.*

Definizione A.17 ***Ascendente, Discendente***
Sia G=($\mathcal{V}$,$\mathcal{A}$) un grafo. I nodi v_i e v_j sono detti adiacenti (v_i,v_j) o $(v_j,v_i) \in \mathcal{A}$.

Definizione A.18 ***Parente, Figlio***
Sia G=($\mathcal{V}$,$\mathcal{A}$) un grafo diretto. Se $(v_i,v_j) \in \mathcal{A}$, allora v_i si dice parente di v_j e, viceversa, v_j si dice figlio di v_i.

L'insieme dei parenti di un nodo v_j si denota con $Pa(v_j)$, l'insieme dei figli di un nodo v_i con $Ch(v_i)$. In questo contesto, si hanno le seguenti caratterizzazioni:

Definizione A.19 ***Foglia, Radice***
Un nodo foglia in un grafo è un nodo che non ha figli. Un nodo radice in un grafo è un nodo che non ha parenti.

Definizione A.20 ***Ascendente, Discendente***
Sia G=($\mathcal{V}$,$\mathcal{A}$) un grafo diretto, e siano v_i e v_j due nodi. Se esiste un cammino da v_i a v_j, allora v_i è un ascendente di v_j, e viceversa, v_j è un discendente di v_i.

I nodi che non sono ascendenti (o discendenti), sono detti nodi non ascendenti (o non discendenti).

Le nozioni della Teoria dei grafi presentate sono sufficienti per la descrizione qualitativa della conoscenza in una rete Bayesiana. Per la descrizione quantitativa della conoscenza, è necessario definire i concetti principali della Teoria della Probabilità.

Appendice B

Richiami di Teoria della Probabilità

Le reti Bayesiane hanno la proprietà di rappresentare l'incertezza attraverso la combinazione di elementi della Teoria dei grafi, che fornisce gli strumenti necessari per una modellizzazione qualitativa della conoscenza, e della Teoria della Probabilità, che permette di quantificare l'incertezza sulle conoscenze.

B.1 Probabilità

Si presentano alcuni richiami di Teoria della Probabilità, limitando la descrizione agli aspetti legati al suo utilizzo nel contesto delle reti Bayesiane. In particolare, lo spazio in cui verranno definite le probabilità è discreto e finito.

B.1.1 Definizioni principali

Siano Ω un insieme finito non vuoto e $\mathcal{F}$ una σ-algebra di parti di Ω. Sulla coppia $(\Omega,\mathcal{F})$, spazio misurabile, viene definita una funzione che probabilizza tutti gli insiemi che appartengono a $\mathcal{F}$.

Definizione B.1 ***Probabilità***
Sia $(\Omega,\mathcal{F})$ uno spazio misurabile. Una funzione $P : \mathcal{F} \rightarrow [0,1]$ è una probabilità su $(\Omega,\mathcal{F})$ se e solamente se si verifica:

1. $\forall A \in \mathcal{F}$, $0 \leq P(A) \leq 1$;
2. $P(\Omega) = 1$ *(e dunque* $P(\emptyset = 0)$*.*
3. $\forall A_1,A_2 \in \mathcal{F}$, $[A_1 \cap A_2 = \emptyset] \Rightarrow P(A_1 \cup A_2) = P(A_1) + P(A_2)$. A_1 *e* A_2 *sono detti mutuamente esclusivi;*

4. $\forall\{A_i\}, i = 1,2,\ldots$*, a due a due mutuamente esclusivi appartenenti a* $\mathcal{F}$*,*

$$P\left(\cup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} P(A_i)$$

La terna $(\Omega,\mathcal{F},P)$ *viene definita spazio di probabilità.*

Un evento su Ω è un sottoinsieme di Ω. Ω è detto *evento certo*, mentre $\emptyset$ è l'*evento impossibile*.

Definizione B.2 ***Variabile aleatoria***
Una variabile aleatoria (v.a.) è una funzione $X : \Omega \rightarrow \mathcal{D}_X$ *tale che*

$$X : \omega \mapsto X(\omega)$$

Per $x \in \mathcal{D}_X$*, si denota* $\{X = x\}$ *l'evento* $\{\omega \in \Omega | X(\omega) = x\}$.
$\mathcal{D}_X$ *è il dominio di definizione di* X *ovvero l'insieme dei possibili valori assunti da* X.

Una v.a. è denotata da una lettera maiuscola (per esempio, X, X_i) e il suo stato, ovvero il valore che essa assume, attraverso la corrispondente lettera minuscola (per esempio, x, x_i). Un insieme di v.a. è denotato da una lettera maiuscola in grassetto (per esempio, $\mathbf{X}$, $\mathbf{Y}$), e la corrispondente lettera minuscola in grassetto (per esempio $\mathbf{x}$, $\mathbf{y}$) denota una assegnazione, o stato, per ogni variabile nel corrispondente insieme.

B.1.2 Probabilità su un insieme di variabili

Definizione B.3 ***Probabilità congiunta***
Siano $(\Omega,\mathcal{F},P)$ *uno spazio di probabilità, e* X *e* Y *due v.a. definite sullo stesso* Ω*. Si definisce la probabilità congiunta* $P_{XY} : \mathcal{D}_X \times \mathcal{D}_Y \rightarrow [0,1]$ *la funzione:*

$$\begin{array}{rcccl} P_{XY} : (x,y) & \mapsto & P_{XY}(x,y) & = & P(\{X = x\} \cap \{Y = y\}) \\ & & & = & P(\{\omega \in \Omega | X(\omega) = x \wedge Y(\omega) = y\}) \end{array}$$

Questa definizione può essere estesa a tutti gli insiemi finiti $\mathbf{X} = \{X_1,\ldots,X_n\}$ *di v.a. definite sullo stesso* Ω*,* $P_{\mathbf{X}} : \bigotimes_{i\in\{1,\ldots,n\}} \mathcal{D}_{X_i} \rightarrow [0,1]$*, tale che*

$$\begin{array}{rcccl} P_{\mathbf{X}} : \mathbf{u} = (x_1,\ldots,x_n) & \mapsto & P_{\mathbf{X}}(\mathbf{u}) & = & P\left(\cap_{i\in\{1,\ldots,n\}}\{X_i = x_i\}\right) \\ & & & = & P\left(\left\{\omega \in \Omega | \wedge_{i\in\{1,\ldots,n\}} X_i(\omega) = x_i\right\}\right) \end{array}$$

Sia $\mathbf{X}$ un insieme finito e non vuoto di v.a. discrete su Ω. $\mathcal{D}_{\mathbf{X}} = \otimes_{X_i \in \mathbf{X}}(\mathcal{D}_{X_i})$, prodotto cartesiano dei domini di definizione delle variabili in $\mathbf{X}$, è lo spazio degli stati di $\mathbf{X}$. Un elemento $\mathbf{x} \in D_{\mathbf{X}}$ è detto una *configurazione* di $\mathbf{X}$.

A partire dalla probabilità congiunta di un insieme di variabili, è possibile calcolare la probabilità di un qualsiasi suo sottoinsieme. Quest'ultima viene detta *probabilità marginale*.

Proprietà B.1 ***Marginalizzazione***
Siano $\mathbf{X}$ un insieme finito, non vuoto, di v.a., $\mathbf{U} \subset \mathbf{X}$ non vuoto, $\mathbf{U}' = \mathbf{X} \setminus \mathbf{U}$ e $P(\mathbf{X})$ la probabilità congiunta sulle variabili di $\mathbf{X}$. Si definisce marginalizzazione di P su $\mathbf{U}$ la funzione:

$$\forall \mathbf{u} \in \mathcal{D}_{\mathbf{U}}, P(\mathbf{u}) = \sum_{\mathbf{u}' \in \mathcal{D}_{\mathbf{U}'}} P(\mathbf{u},\mathbf{u}')$$

Questa funzione corrisponde alla probabilità congiunta delle variabili di $\mathbf{U}$.

L'operazione di marginalizzazione può essere generalizzata a qualsiasi funzione f su un insieme di varibili $\mathbf{X}$. La notazione usuale (Jensen, 1996) per questa operazione è $[f]^{\downarrow \mathbf{U}}$ dove $\mathbf{U} \subset \mathbf{X}$. Dunque la proprietà precedente può essere riscritta nel seguente modo:

$$\forall \mathbf{U} \subset \mathbf{X}, P(\mathbf{U}) = [P(\mathbf{X})]^{\downarrow \mathbf{U}} = \sum_{\mathbf{u}' \in \mathbf{X} \setminus \mathbf{U}} P(\mathbf{U},\mathbf{u}')$$

Si richiamano alcuni concetti, che permettono di tenere in considerazione l'informazione acquisita.

Definizione B.4 ***Probabilità condizionata***
Siano X e Y due varibili aleatorie su $(\Omega,\mathcal{F},P)$. Per ogni $x \in \mathcal{D}_X$ e $y \in \mathcal{D}_Y$, la probabilità condizionata di $X = x$ dato $Y = y$ è il valore $P(X = x|Y = y)$ che verifica:

$$P(X = x,Y = y) = P(X = x|Y = y)P(Y = y)$$

o, in generale, la probabilità condizionata di X, dato $Y = y$, verifica:

$$P(X,Y = y) = P(X|Y = y)P(Y = y)$$

Questa definizione si generalizza nella seguente definizione:

Definizione B.5 ***Probabilità condizionata generalizzata***
Sia $(X_i)_{i \in \{1,\ldots,n\}}$ un insieme di v.a. sullo stesso Ω. Allora:

$$\begin{aligned} P(x_1,\ldots,x_n) &= P(x_1)P(x_2,\ldots,x_n|x_1) \\ &= P(x_1)P(x_2|x_1)P(x_3,\ldots,x_n|x_1,x_2) \\ &= \textstyle\prod_{i=1}^{n} P(x_i|x_1,\ldots,x_{i-1}) \end{aligned}$$

Si utilizza, talvolta, la convenzione $P(X|\emptyset) = P(X)$.

Questa definizione conduce al *Teorema di Bayes*:

Teorema B.1 ***di Bayes***
Se $P(Y = y)$ è positiva, allora

$$P(X = x|Y = y) = \frac{P(Y = y|X = x)P(X = x)}{P(Y = y)}.$$

Più in generale, si ha:

$$P(X = x|Y = y,Z = z) = \frac{P(Y = y|X = x,Z = z)P(X = x)}{P(Y = y|Z = z)}$$

B.2 Indipendenza condizionata

Il calcolo e l'uso della probabilità congiunta su un insieme di variabili, ha complessità esponenziale nel numero delle variabili coinvolte. Per rendere trattabili i calcoli su questa quantità, è necessario ridurne la complessità: ciò è possibile considerando la nozione di *indipendenza condizionata*.

Definizione B.6 ***Indipendenza condizionata***
Siano $X,Y,Z \in \mathbf{X}$ variabili casuali, o sottoinsiemi di variabili, su Ω. X e Y sono condizionatamente indipendenti dato Z se $P(X = x|Y = y,Z = z) = P(X = x|Z = z)$, o equivalentemente $P(X = x,Y = y|Z = z) = P(X = x|Z = z) \cdot P(Y = y|Z = z)$, per ogni valore x, y e z che le variabili possono assumere, .

L'indipendenza condizionata viene indicata con l'espressione $X \perp\!\!\!\perp Y|Z$. Si noti che la indipendenza non condizionata (indipendenza marginale) può essere trattata come un caso particolare della indipendenza condizionata considerando X, Y e $\emptyset$, in modo tale che
$X \perp\!\!\!\perp Y \mid \emptyset$.

La definizione implica che la conoscenza sul valore assunto da Y non apporta alcuna informazione per la conoscenza di X, quando il valore di Z è noto. L'indipendenza condizionata è in relazione con le distribuzioni di probabilità condizionate in modo tale che:

$$\begin{aligned} X \perp\!\!\!\perp Y|Z &\iff P(X|Y,Z) = P(X|Z) \\ &\iff P(X,Y|Z) = P(X|Z)P(Y|Z) \\ &\iff P(X,Y,Z) = P(X|Z)P(Y|Z)P(Z) \end{aligned}$$

Ne segue che una probabilità congiunta $P(\mathbf{X}) = P(X_1,\ldots,X_n)$, fattorizzabile in probabilità condizionate

$$P(X_1,\ldots,X_n) = \prod_{i=1}^{n} P(X_i|X_1,\ldots,X_{i-1}),$$

può essere fattorizzata ulteriormente attraverso le indipendenze condizionate rilevate. Si ha il seguente:

Teorema B.2
Per ogni i, sia $\mathbf{U}_i \subset \{X_1, \ldots, X_{i-1}\}$ tale che $X_i \perp\!\!\!\perp (\{X_1, \ldots, X_{i-1}\} \setminus \mathbf{U}_i) | \mathbf{U}_i$, allora:

$$P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i | \mathbf{U}_i).$$

B.2.1 Proprietà

L'indipendenza condizionata verifica un insieme di proprietà; fra queste, si richiamano le seguenti. Siano X, Y, Z e W variabili aleatorie:

Se $X \perp\!\!\!\perp Y \mid Z$	allora $Y \perp\!\!\!\perp X \mid Z$
Se $X \perp\!\!\!\perp Y \mid Z$ e $\exists f$ tale che $U = f(X)$	allora $U \perp\!\!\!\perp Y \mid Z$
Se $X \perp\!\!\!\perp Y \mid Z$ e $\exists f$ tale che $U = f(X)$	allora $X \perp\!\!\!\perp Y \mid Z,U$
Se $X \perp\!\!\!\perp Y \mid Z$ e $X \perp\!\!\!\perp W \mid Y,Z$	allora $X \perp\!\!\!\perp Y,W \mid Z$

Appendice C

Software

Il rapido sviluppo della ricerca sulle reti Bayesiane ha portato negli ultimi 15 anni ad una veloce crescita del numero dei software legati a questo strumento, sia a scopo di ricerca sia per scopi puramente applicativi. Questa appendice ha lo scopo di presentare alcuni dei software presenti in letteratura.

La lista dei software compilata da Kevin Murphy è una buona base di partenza per avere una panoramica delle caratteristiche legate ad ognuno dei software presenti in elenco. E' possibile trovare questa lista all'indirizzo web:

`http://www.ai.mit.edu/ murphyk/Software/BNT/bnsoft.html`.

In FiguraC.1 si evidenziano le caratteristiche base dei software in elenco, come ad esempio informazioni tecniche sulle risorse disponibili, la possibilità di usare un'interfaccia grafica, il tipo di nodi analizzati e altre funzionalità. E' inoltre evidenziato se il software contiene metodologie di apprendimento dei parametri o della struttura e il tipo di inferenza probabilistica supportata. I dettagli del significato di ogni colonna sono descritti in Figura C.2. Queste due Figure sono tratte da Korb e Nicholson (2004).

Inoltre, è disponibile la lista degli strumenti di Google all'indirizzo web:

`http://directory.google.com/Top/Computers/Artificial_Intelligence/Belief_Networks/Software/`

mentre la Bayesian Network Repository, contenente esempi di reti Bayesiane e i corrispondenti database, si trova in :

`http://www.cs.huji.ac.uil/labs/combio/Repository`

Si descrivono, in seguito, alcuni tra i principali software legati alle reti Bayesiane.

Src Is the source code included? **N=no.** If yes, what language? **J** = Java, **M** = Matlab, **L** = Lisp, **C**, **C++**, **R**, **A** = APL.

API Is an application program interface included?

N means the program cannot be integrated into your code, i.e., it must be run as a standalone executable. **Y** means it can be integrated.

Exec The **executable** runs on: **W** = Windows (95/98/2000/NT), **U** = Unix, **M** = Mac-Intosh, **-** = Any machine with a compiler.

GUI Is a Graphical User Interface included? **Y=Yes,N=No.**

D/C Are continuous-valued nodes supported (as well as discrete)? **G** = (condition-ally) Gaussians nodes supported analytically, **Cs** = continuous nodes supported by sampling, **Cd** = continuous nodes supported by discretization, **Cx** = con-tinuous nodes supported by some unspecified method, **D** = only discrete nodes supported.

DN Are decision networks/influence diagrams supported? **Y=Yes,N=No.**

Params Does the software functionality include parameter learning? **Y=Yes,N=No.**

Struct Does the software functionality include structure learning? **Y=Yes,N=No.**

CI means **Y**, using conditional independency tests

K2 means **Y**, using Cooper & Herskovits' K2 algorithm

D/U What kind of graphs are supported? **U** = only undirected graphs, **D** = only directed graphs, **UD** = both undirected and directed, **CG** = chain graphs (mixed directed/undirected).

Inf Which inference algorithm is used?

JT = Junction Tree, **VE = variable (bucket) elimination**, **PT = Pearl's poly-tree**, **E = Exact inference** (unspecified), **MH = Metropolis Hastings**, **MC = Markov chain Monte Carlo (MCMC)**, **GS = Gibbs sampling**, **IS = Impor-tance sampling**, **S = Sampling**, **O = Other** (usually special purpose), **++ = Many** methods provided, **?** = Not specified, **N** = None, the program is only designed for structure learning from completely observed data.

NB: Some packages support a form of sampling (e.g., **likelihood weighting**, **MDMC**), in addition to their exact algorithm; this is indicated by **(+S)**.

Free Is a free version available? **O**=Free (though possibly only for academic use), **$** = Commercial (although most have free versions which are restricted in various ways, e.g., the model size is limited, or models cannot be saved, or there is no API.)

Figura C.1. Descrizione delle caratteristiche presentate per i software - Lista di Kevin Murphy

-

Name	Src	API	Exec	GUI	D/C	DN	Params	Struct	D/U	Infer	Free
Analytica	N	Y	WM	Y	G	Y	N	N	D	S	$
Bassist	C++	Y	U	N	G	N	Y	N	D	MH	O
Bayda	J	Y	WUM	Y	G	N	Y	N	D	?	O
BayesBuilder	N	N	W	Y	D	N	N	N	D	?	O
BayesiaLab	N	N	-	Y	Cd	N	Y	Y	CG	JT,G	$
Bayesware	N	N	WUM	Y	Cd	N	Y	Y	D	?	$
B-course	N	N	WUM	Y	Cd	N	Y	Y	D	?	O
BNPC	N	Y	W	Y	D	N	Y	CI	D	?	O
BNT	M/C	Y	WUM	N	G	Y	Y	Y	UD	S,E(++)	O
BNJ	J	Y	-	Y	D	N	N	Y	D	JT,IS	O
BucketElim	C++	Y	WU	N	D	N	N	N	D	VE	O
BUGS	N	N	WU	Y	Cs	N	Y	N	D	GS	O
BusNav	N	N	W	Y	Cd	N	Y	Y	D	JT	$
CABeN	C	Y	WU	N	D	N	N	N	D	S(++)	O
CaMML	N	N	U	N	Cx	N	Y	Y	D	N	O
CoCo+Xlisp	C/L	Y	U	Y	D	N	Y	CI	U	JT	O
CIspace	J	N	WU	Y	D	N	N	N	D	VE	O
Deal	R	-	-	Y	G	N	N	Y	D	N	O
Ergo	N	Y	WM	Y	D	N	N	N	D	JT(+S)	$
First Bayes	A	N	W	Y	-	N	N	N	–	O	O
GDAGsim	C	Y	WUM	N	G	N	N	N	D	E	O
GeNIe/SMILE	N	Y	WU	Y	D	Y	N	N	D	JT(+S)	O
GMRFsim	C	Y	WUM	N	G	N	N	N	U	MC	O
GMTk	N	Y	U	N	D	N	Y	Y	D	JT	O
gR	R	-	-	-	-	-	-	-	-	-	O
Grappa	R	Y	-	N	D	N	N	N	D	JT	O
Hugin	N	Y	WU	Y	G	Y	Y	CI	CG	JT	$
Hydra	J	Y	-	Y	Cs	N	Y	N	UD	MC	O
Ideal	L	Y	WUM	Y	D	Y	N	N	D	JT	0
Java Bayes	J	Y	WUM	Y	D	Y	N	N	D	JT,VE	O
MIM	N	N	W	Y	G	N	Y	Y	CG	JT	$
MSBNx	N	Y	W	Y	D	Y	N	N	D	JT	O
Netica	N	Y	WUM	Y	G	Y	Y	N	D	JT	$
PMT	M/C	Y	-	N	D	N	Y	N	D	O	O
PNL	C++	Y	-	N	D	N	Y	Y	UD	JT	O
Pulcinella	L	Y	WUM	Y	D	N	N	N	D	?	O
RISO	J	Y	WUM	Y	G	N	N	N	D	PT	O
TETRAD IV	N	N	WU	Y	Cx	N	Y	CI	UD	N	O
UnBBayes	J	Y	-	Y	D	N	N	Y	D	JT	O
Vibes	J	Y	WU	Y	Cx	N	Y	N	D	?	O
Web Weaver	J	Y	WUM	Y	D	Y	N	N	D	?	O
WinMine	N	N	W	Y	Cx	N	Y	Y	UD	N	O
XBAIES 2.0	N	N	W	Y	G	Y	Y	Y	CG	JT	O

Figura C.2. Confronto delle caratteristiche dei software - Lista di Kevin Murphy -

C.1 Analytica

Lumina Decision Systems, Inc.
26010 Highland Way, Los gatos, CA 95033
`http://www.lumina.com`

Lumina Decision System. Inc., fu fondata nel 1991 da Max Henrion e Brian Arnold. La caratteristica di Analytica è di usare i diagrammi di influenza come strumento statistico di supporto decisionale. Analytica non usa la terminologia dellle reti Bayesiane, comportando una difficoltà nell'identificazione degli aspetti della sua funzionalità.

Analytica 2.0 GUI è disponibile per Windows e Macinthos; il suo API (detto Analytica Decision Engine) è disponibile per Windows 95/98 o NT 4.0.

Il software supporta l'analisi di molte distribuzioni discrete e continue e fornisce un ampio numero di funzioni matematiche e statistiche. L'inferenza è basata su metodi approssimati, e la versione GUI fornisce molti modi per vedere i risultati dell'inferenza, sia attravesro l'uso di grafici che di tabelle. L'evidenza considerata è di tipo hard, e sono presenti molti strumenti per la valutazione dei risultati ottenuti, in particolare ciò che viene chiamato Analisi di Sensibilità.

Analytica non presenta alcuna tecnica di apprendimento, ma nasce con lo scopo di fornire uno strumento per l'analisi del sistema oggetto di studio basata su reti Bayesiane, una volta che la selezione del modello è stata effettuata.

C.2 BayesiaLab

BAYESIA
6, rue Lonard de Vinci - BP0102, 53001 Laval Cedex, France
`htpp://www.bayesia.com`

BayesiaLab GUI è disponibile per tutti i sistemi operativi che supportano JRE. Esiste inoltre un prodotto, "BEST", che usa le reti Bayesiane per la diagnosi e la riparazione.

Le variabili continue devono essere discretizzate. Quando si definiscono le variabili a partire da un database, BayesiaLab supporta la discretizzazione in intervalli di uguale distanza o uguale frequenza, oppure attraverso un approccio basato sugli alberi di decisione, in modo tale che gli intervalli scelti dipendano dall'informazione che essi apportano ad una specifica variabile obiettivo.

L'inferenza è basata per default sul metodo della Junction Tree Propagation, ma è possibile usare un algoritmo di inferenza approssimata basato su metodi MCMC.

BayesiaLab apprende i parametri della rete attraverso le tecniche Bayesiane fornendo stime di tipo MAP. Fornisce tre metodologie per l'apprendimento della struttura: un approccio detto SopLEQ, che usa le proprietà di equivalenza delle reti Bayesiane e due versioni dell'approccio Taboo Search.

Fornisce inoltre la possibilità di trattare con dati mancanti.

C.3 Bayes Net Toolbox (BNT)

Kevin Murphy
MIT AI lab, #200 Technology Square, Cambridge, MA 02139
`http://www.ai.mit.edu/~murphyk/Software/BNT/bnt.html`

Questo strumento, supportato in MATLAB, non ha una versione GUI ed è distribuito con licenza GNU (Library General Public).

Le variabili analizzate possono essere continue o discrete, e i metodi di inferenza si basano sia su algoritmi esatti che approssimati.

L'apprendimento dei parametri fornisce stime basate sia sulla verosimiglianza (MLE) che sull'approccio Bayesiano (MAP). Nel caso di dati incompleti, utilizza il metodo EM. L'apprendimento della struttura usa algoritmi di tipo greedy Hill-Climbing con funzioni score di tipo Bayesiano (solo nel caso di dati completi) e permette anche l'approccio Constraint-based attraverso gli algoritmi IC e PC nel caso di dati completi e gli algoritmi IC* e FCI nel caso di dati incompleti.

C.4 Bayesware Discoverer

Bayesware Ltd.
`http://www.bayesware.com`

Bayesware Discoverer è una software ad interfaccia grafica per Windows. E' la versione commerciale del Bayesian Knowledge Discoverer, sviluppato al Knowledge Media Institute della Open University (UK) e basato sulle ricerche di Marco Ramoni e Paola Sebastiani.

Il software supporta l'analisi di variabili discrete, e discretizza le variabili continue in

intervalli ad uguale frequenza o dimensione. Permette inoltre di trattarre il caso di dati mancanti.

La procedura per la stima delle distribuzioni di probabilità condizionate è basata sull'approccio Bayesiano. Nel caso di caso di dati incompleti si utilizza la proposta di Ramoni e Sebastiani (1998).

Per l'apprendimento della struttura, si considera l'algoritmo K2 nel caso di dati completi, mentre viene utilizzata la procedura proposta da Ramoni e Sebastiani (1997) nel caso di database incompleto.

Lo strumento implementa una versione leggermente modificata dell'algoritmo proposto da Shatcher (1990) per la propagazione dell'evidenza.

C.5 Elvira

Elvira Consortium
`http://www.ual.es/personal/asalmero/elvira/presenta01.swf`

Elvira[1] nasce nel 1997 dall'unione di due progetti sui modelli grafici probabilistici, in particolare sulle reti Bayesiane, supportati dal Ministero Spagnolo per la Scienza e la Tecnologia, con lo scopo iniziale di fornire uno strumento in grado di unire le procedure sviluppate da ognuno dei due gruppi di ricercatori coinvolti. Dal 2004, lo scopo è quello di costruire un sistema in grado di analizzare attraverso la metodologia fornita dalle reti Bayesiane sistemi complessi reali.

Elvira ha un'interfaccia grafica, che non mantiene però le potenzialità della versione API, disponibile in linguaggio JAVA.

E' in grado di analizzare sia variabili continue che discrete, anche se le sue potenzialità si riferiscono principalmente al caso discreto.

Sono implementati molti metodi per l'inferenza probabilistica, sia esatta che approssimata. Per quanto riguarda la propagazione esatta, si considerano, fra gli altri, la Junction Tree Propagation e l'algoritmo Variable Elimination.

I parametri possono essere appresi sia attraverso metodi di stima di massima verosimiglianza che metodi Bayesiani.

Per l'apprendimento della struttura, Elvira implementa l'algoritmo PC, e metodi greedy Hill-Climbing basati sulle funzioni score BIC, BDe, e l'algoritmo K2. Quest'ultimo viene modificato in un algoritmo chiamato K2SN, in cui cade l'assunzione di ordinamento delle

[1] Elvira non è presente nella Figura riassuntiva dei software (Figura C.2) in quanto la sua divulgazione è recente.

variabili. E' inoltre annunciato l'inserimento di un algoritmo che ricerca nello spazio dei DAG equivalenti. Esistono altri algoritmi implementati nel sistema, non descritti all'interno della tesi, che non vengono dunque citati.

Elvira offre inoltre alcuni dei più utilizzati metodi per la valutazione delle reti Bayesiane. A questo proposito, si veda Lacave e Diez (2002).

C.6 Hugin

Hugin Expert, Ltd.
Niels Jernes Vej 10,9220 Aalborg East, Denmark
`http://www.hugin.com`

Inizialmente Hugin fu sviluppato da un gruppo di ricercatori della Aalborg University, continuato da Steffen L. Lauritzen e Finn V. Jensen, finchè non venne deciso di rendere commerciale il prodotto attraverso il gruppo chiamato Hugin Expert. Nonostante ciò, la stretta collaborazione tra Hugin Expert e l'università di Aalborg permette di tenere in costante aggiornamento il software alle nuove ricerche fatte nell'ambito delle reti Bayesiane. Per questo, tuttora, Hugin risulta essere uno degli strumenti più efficienti e più utilizzati nell'ambito dello studio e della ricerca legata alle reti Bayesiane.

Hugin GUI è disponibile per i sistemi operativi Sun Solaris, Windows e Linux; Hugin API, detto anche "Hugin Decision Engine" è disponibile, tra gli altri, in linguaggio C++ e Java.

Le variabili analizzate possono essere continue, discrete o di entrambi i tipi contemporaneamente.

L'algoritmo base per l'inferenza probabilistica è il Junction Tree Propagation ed è possibile la visualizzazione del JT costruito. Esiste la possibilità di sciegliere il metodo di triangolarizzazione da utilizzare così come la possibilità di utilizzare una versione approssimata dell'algoritmo.

L'apprendimento dei parametri è fatto utilizzando l'algoritmo EM, mentre l'apprendimento della struttura supporta solo l'algoritmo PC.

C.7 Netica

Norsys Software Corp.
3512 W 23^{rd} Ave., Vancouver, BC, Canada V6S 1K5

`http://www.norsys.com`

Lo sviluppo di Netica è iniziato nel 1992 dal lavoro di Brent Boerlage, mentre la sua commercializzazione è avvenuta successivamente nel 1995.

Netica GUI è disponibile per i sistemi operativi Windows e Mac, mentre Netica API è disponibile nei linguaggi C e JAVA.

Le variabili analizzate possono essere determinate direttamente da un database e possono essere sia discrete che continue (quest'ultime sono discretizzate in intervalli).

Il processo inferenziale è basato sulla Junction Tree Propagation. La determinazione del JT avviene utilizzando tecniche che minimizzano la dimensione delle clique del grafo. E' possibile visualizzare il JT e la sequenza di eliminazione determinata.

Netica supporta solo l'apprendimento dei parametri. Nel caso di dati completi, si utilizza l'approccio Bayesiano, permettendo di specificare probabilità a priori. Nel caso di dati incompleti è usato l'algoritmo EM.

Fornisce inoltre la possibilità di simulare database a partire da una rete Bayesiana specificata, e supporta numerosi metodi per la valutazione delle reti, ad esempio, analisi della sensitività e misure di convalida statistica come l'accuratezza predittiva di una variabile obiettivo selezionata o la matrice di confusione.

C.8 TETRAD

Peter Spirtes, Clark Glymour e Richard Scheines
Dept. of Philosophy, Carnegie Mellon University
`http://www.phil.cmu.edu/tetrad/`

L'ultima versione è TETRAD IV, successore del primo programma s, TETRAD II, sviluppato a partire dagli studi sull'apprendimento strutturale condotti da P. Spirtes, C. Glymour e R. Scheines (Spirtes et al., 1993) basati sull'approccio Constraint-based. Il software ha un'interfaccia grafica supportata da JRE.

L'apprendimento dei parametri utilizza le tecniche di stima basate sulla massima verosimiglianza, mentre gli algoritmi utilizzati per l'apprendimento della struttura sono l'algoritmo PC, l'algoritmo FCI (nel caso di dati mancanti) e una ricerca basata su un algoritmo genetico.

Bibliografia

Akaike, H. (1970) Statistical Predictor Identification. *Ann. Inst. Statist. Math.*, **22**: 203-217.

Beinlich, I., Suermondt,H., Chavez, R., Cooper, G. (1989) The Alarm Monitoring system: a Case Study with two Probabilistic Inference Techniques for Belief Networks. In *Proceedings Artificial Intelligence in Medical Care*, 247-256.

Berge, C. (1958) *Theorie des Graphes et ses Applications*. Dunod.

Berge, C. (1973) *Graphs and Hypergaphs*. North-Holland, Amsterdam.

Beygelzimer, A., Rish, I. (2002) Inference Complexity as a Model-Selection Criterion for Learning Bayesian Networks. In *Proceedings of the Eighth International Conference on Principles of Knowledge Representation and Reasoninig*, 558-567.

Bolzan, M., Brogini, A., Slanzi D. (2005) Apprendimento di Modelli Grafici Esplorativi per la Valutazione in Ambito Socio-Sanitario: il Caso dell'Assistenza Informale. *Non Profit* **1**: 207-224.

Bouckaert, R.R. (1995) *Bayesian Belief Networks: from Construction to Inference*. Ph.D. Thesis. University of Utrecht.

Brogini, A., Bolzan, M., Slanzi D. (2004) Identifying A Bayesian Network For The Problem Hospital And Families: The Analysis Of Patient Satisfaction With Their Stay In Hospital.In *Applied Bayesian Statistical Studies in Biology and Medicine*. Eds. M. di Bacco et al. Kluwer Academic Publisher, Cap.4.

Brogini, A., Slanzi, D. (2005) Unsupervised Vs Supervised Learning In A Real Complex System. Atti del Convegno: *Sistemi Complessi e Statistica Computazionale -S.Co. 2005* -: 467-472, Cleup.

Buntine, W. (1996) A Guide to the Literature on Learning Probabilistic Networks from Data. *IEEE Transactions on Knowledge and Data Engineering*,**8(2)**: 195-210.

Buntine, W. (1991) Theory Refinement on Bayesian Networks. In *Proceeding of the Seventh Conference on Uncertainty in Artificial Intelligence*, 52-60.

Castillo, E., Gutierrez, J.M., Hadi, A.S. (1997) *Expert Systems and Probabilistic Network Models*., Springer-Verlag.

Cheng, J., Bell, D.A., Liu, W. (1997) Learning Belief Networks from Data: an Information Theory Based Approach. In *Proceeding of the Sixth International Conference on Information and Knowledge Management*, 225-331.

Chickering, D. (1996) Learning Equivalence Classes of Bayesian Network Structures. In *Proceeding of the Twelfth Conference on Uncertainty in Artificial Intelligence*, 150-157.

Chickering, D. (2002) Optimal Structure Identification with Greedy Search. *Journal of Machine Learning Research*, **3**: 507-554.

Chickering, D., Heckerman, D. (1997) Efficient Approximations for the Marginal Likelihood of Incomplete Data given a Bayesian Network. *Machine Learning*, **29**:181-212.

Chib, S. (1995) Marginal Likelihood from the Gibbs Output. *Journal of the American Statistical Association*, **90**: 1313-1321.

Chow, C.K., Liu, C.N. (1968b) Approximating Discrete Probability Distributions with Dependence Trees. *IEEE Transactions on information Theory*, **14(3)**: 462-467.

Cooper, G. F. (1990) The Computational Complexity of Probabilistic Inference using Bayesian Belief Networks. *Artificial Intelligence*, **42**: 393-405.

Cooper, G.F. (1995a) Causal Discovery from Data in the Presence of Selection Bias. In *Proceedings of the Fifth International Workshop on Artificial Intelligence and Statistics*. Fort Lauderdale, Florida.

Cooper, G.F., Herskovits, E. (1992) A Bayesian Method for the Induction of Probabilistic Networks from Data. *Machine Learning*, **9(4)**: 309-348.

Cowell, R.G., Dawid, A.P., Lauritzen, S.L., Spieghelhalter, D.J. (1999) *Probabilistic Networks and Expert Systems*. Springer-Verlag.

Dagum P., Horvitz E. (1993) Approximating Probabilistic Inference in Bayesian Belief Networks is NP-hard. *Artificial Intelligence*, **60**: 141-153.

de Campos, L.M. (1998) Independency Relationships and Learning Algorithms for Singly Connected Networks. *Journal of Experimental and Theoretical Artificial Intelligence*, **10(4)**: 511-549.

Dempster, A.P., Laird, N.M., Rubin, D.B. (1977) Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society, Series B*, **39**:1-38.

Flores M. J., Gamez J. A., Olesen K. G. (2003) Incremental Compilation of Bayesian Networks. *Proceeding of the Conference in Artificial Intelligence*, 233-240.

Friedman, N. (1998) The Bayesian Structural EM. In *Proceedings of the Fourteenth Conference in Uncertainty in Artificial Intelligence*.

Friedman, N., Geiger, D., Goldszmidt, M. (1997) Bayesian Network Classifiers. *Machine Learning*, **29(2-3)**: 131-163.

Geiger, D., Pearl, J. (1990) On the logic of Causal Models. In *Proceedings of the Fourth Conference in Uncertainty in Artificial Intelligence*: 136-147.

Geiger, D., Verma, T., Pearl, J. (1990) d-separation: from Theorems to Algorithms. In *Proceedings of the Fifth Conference in Uncertainty in Artificial Intelligence*.

Glymour, C., Cooper, G.F. (Eds). (1999) *Computation, Causation and Discovery*. AAAI Press.

Golumbic, M.C. (1980) Triangulated Graphs. *Algorithmic Graph Theory and Perfect Graphs*, 98-100.

Heckerman, D. (1996) A Tutorial on Learning with Bayesian Networks. Technical Report # MSR-TR-95-06, Microsoft Research, Redmond, Washington.

Heckerman, D., Geiger, D., Chickering, D.M. (1995) Learning Bayesian Networks: the Combinations of knowledge and Statistical Data. *Machine Learning*, **20**:197-243.

Hsu, W., Guo, H., Perry, B., Stilson J. (2002) A Permutation Genetic Algorithm for Variable Ordering in Learning Bayesian Networks from Data. In *GECCO 2002: Proceedings of the Genetic and Evolutionary Computation Conference*, 383-390.

Huang, C., Darwiche, A. (1996) Inference in Belief Networks: a Procedural Guide. *International Journal of Approximate Reasoning*, **15**: 225-263.

Jensen, F.V. (2001) *Bayesian Networks and Decision Graphs*. Springer-Verlag.

Jensen, F.V. (1996) *An Introduction to Bayesian Networks*. Springer-Verlag.

Jensen, F. V., Lauritzen, S. L., Olesen, K. G. (1990) Bayesian Updating in Causal Probabilistic Networks by Local Computations. *Computational Statistics Quarterly* **4**: 269-282.

Jensen, F.V., Jensen, F. (1994) Optimal Junction Trees. *Proceedings of the Tenth Conference on Uncertainty in Artificial Intelligence*, 360-366.

Kiærulff, U. (1990) Triangulation of Graphs - Algorithms Giving Small total State Space. Th. Report R-90-09, Dept. of Math. and Comp. Sci., Aalborg University, Denmark.

Korb, K.B., Nicholson, A.E. (2004) *Bayesian Artificial Intelligence*. Chapman & Hall/CRC.

Lacave, C., Diez, F.J. (2002) Explanation for Causal Bayesian Networks in Elvira. In *Proceedings of the Whorkshop on Intelligent Data Analysis in Medicine and Pharmacology (IDAMAP - 2002)*.

Lam, W., Bacchus, F. (1994) Learning Bayesian Belief Networks. An Approach Based on the MDL Principle. *Computational Intelligence*, **10(4)**: 269-293.

Larrañaga, P., Poza, M., Yurramendi, Y., Murga R.H., Kuijpers, C.M.H. (1996) Structure Learning of Bayesian Networks by Genetic Algorithms: a Performance Analysis of Control Parameters. *IEEE Journal on Pattern Analysis and Machine Intelligence*, **18(9)**: 912-926.

Lauritzen, S.L. (1996) *Graphical Models*. Clarendon Press, Oxford.

Lauritzen, S.L., Dawid, A.P., Larsen, B.N., Leimer, H.G. (1990) Independence Properties of Directed Markov Fields. *Networks*, **20**:491-505.

Lauritzen, S.L., Spiegelhalter, D.J. (1988) Local Computation with Probabilities on Graphical Structures and their Applications to Expert Systems (with dicussion). *Journal of the Royal Statistical Society B*, **50**:157-224.

Margaritis, D. (2003) *Learning Bayesian Network Model Structure from Data*. Ph.D. Thesis. Carnegie Mellon University.

McLachlan, G.J., Krishnan, T. (1997) *The EM Algorithm and its Extensions*. Wiley.

Meek, C. (1997) *Graphical Models: Selecting Causal and Statistical Models*. Ph.D. Thesis. Carnegie Mellon University.

Naïm, P., Wuillemin, P.H. Leray, P., Pourret, O., Becker, A. (2004) *Reseaux Bayesiens*. Editions Eyrolles, Paris.

Neapolitan, R.E. (2004) *Learning Bayesian Networks*. Prentice Hall.

Nishii, R. (1988) Maximum Likelihood Principle and Model Selection when the True Model is Unspecified. *Journal of Multivariate Analysis*, **27**: 392-403.

Pearl, J. (1982) Reverend Bayes on Inference Engines: a Distributed Hierarchical Approach. In *Proceedings of the Second National Conference on Artificial Intelligence*: 133-136.

Pearl, J. (1988) *Probabilistic Reasoning in Intelligence Systems*. Morgan Kaufmann.

Pearl, J. (2000) *Causality: Models, Reasoning, and Inference*. Cambridge University Press.

Pearl, J., Verma T. (1991) A Thery of Inferred Causation. In: Allen, J., Fikes, R., Sandewall, E. (Eds.): *KR'91: Principles of Knowledge Representation and Reasoning*, San Mateo,California, Morgan Kaufmann, 441-452.

Puerta Callejon, J.M. (2001) *Métodos Locales y Distribuidos para la Construccíon de Redes de Creencia Estáticas y Dinámicas*. Ph.D.Thesis. E.T.S. de Ingeniería Informática, Granada.

Ramoni M., Sebastiani, P. (1997) Learning Bayesian Networks from Incomplete Databases. In *Proceedings of the Thirteenth Conference on Uncertainty in Artificial Intelligence*, 401-408.

Ramoni M., Sebastiani, P. (1998) Parameter Estimation in Bayesian Networks from Incomplete Database. *Intelligence Data Analysis*, **2**: 139-160.

Rissanen, J. (1978) Modelling by Shortest Data Description. *Automatica*, **14**: 465-471.

Robinson, R. (1977) Counting Unlabeled Acyclic Digraphs. In: Little, C.: (Ed.) *Lecture Notes in Mathematics - Combinatorial Mathematics V*, Berlin, Springer, 28-43.

Shachter, R.D. (1986) Evaluating Influence Diagrams. *Operation Research*, **34**: 871-882.

Spirtes, S., Glymour, C., Scheines, R. (1993) *Causation, Prediction and Search*. Springer-Verlag.

Spirtes, P., Meek, C., Richardson T. (1995) Causal Inference in the Presence of Latent Variables and Selection Bias. In *Proceedings of the Eleventh Conference in Uncertainty in Artificial Intelligence*.

Schwarz, G. (1978) Estimating the Dimension of a Model. *Annals of Statistics*, **6**: 461-464.

Verma, T., Pearl, J. (1988) Causal Networks: Semantics and Expressiveness. In *Proceedings of the Fourth Workshop on Uncertainty in Artificial Intelligence*, 352-359.

Verma, T., Pearl, J. (1991) Equivalence and Synthesis of Causal Models. In *Proceedings of the Sixth Conference in Uncertainty in Artificial Intelligence*, 352-359.

Wen, W.X. (1991) Optimal Decomposition of Belief Networks. In *Proceedings of the Sixth Conference in Uncertainty in Artificial Intelligence*, 209-224.

Zhang, N.L., Poole, D. (1996) Exploiting Causal Independence in Bayesian Network Inference. In *Journal of Artificial Intelligence Research*, **5**: 301-328.