Learning Bayesian Networks

Marco F. Ramoni
Harvard Medical School, USA

Paola Sebastiani
Boston University School of Public Health, USA

INTRODUCTION

Born at the intersection of artificial intelligence, statistics, and probability, Bayesian networks (Pearl, 1988) are a representation formalism at the cutting edge of knowledge discovery and data mining (Heckerman, 1997). Bayesian networks belong to a more general class of models called *probabilistic graphical models* (Whittaker, 1990; Lauritzen, 1996) that arise from the combination of graph theory and probability theory, and their success rests on their ability to handle complex probabilistic models by decomposing them into smaller, amenable components. A probabilistic graphical model is defined by a graph, where nodes represent stochastic variables and arcs represent dependencies among such variables. These arcs are annotated by probability distribution shaping the interaction between the linked variables. A probabilistic graphical model is called a Bayesian network, when the graph connecting its variables is a directed acyclic graph (DAG). This graph represents conditional independence assumptions that are used to factorize the joint probability distribution of the network variables, thus making the process of learning from a large database amenable to computations. A Bayesian network induced from data can be used to investigate distant relationships between variables, as well as making prediction and explanation, by computing the conditional probability distribution of one variable, given the values of some others.

BACKGROUND

The origins of Bayesian networks can be traced back as far as the early decades of the 20th century, when Sewell Wright developed path analysis to aid the study of genetic inheritance (Wright, 1923, 1934). In their current form, Bayesian networks were introduced in the early 1980s as a knowledge representation formalism to encode and use the information acquired from human experts in automated reasoning systems in order to perform diagnostic, predictive, and explanatory tasks (Charniak, 1991; Pearl, 1986, 1988). Their intuitive graphical nature and their principled probabilistic foundations were very attractive features to acquire and represent information burdened by uncertainty. The development of amenable algorithms to propagate probabilistic information through the graph (Lauritzen, 1988; Pearl, 1988) put Bayesian networks at the forefront of artificial intelligence research. Around the same time, the machine-learning community came to the realization that the sound probabilistic nature of Bayesian networks provided straightforward ways to learn them from data. As Bayesian networks encode assumptions of conditional independence, the first machine-learning approaches to Bayesian networks consisted of searching for conditional independence structures in the data and encoding them as a Bayesian network (Glymour, 1987; Pearl, 1988). Shortly thereafter, Cooper and Herskovitz (1992) introduced a Bayesian method that was further refined by Heckerman, et al. (1995) to learn Bayesian networks from data.

These results spurred the interest of the data-mining and knowledge-discovery community in the unique features of Bayesian networks (Heckerman, 1997); that is, a highly symbolic formalism, originally developed to be used and understood by humans, well-grounded on the sound foundations of statistics and probability theory, able to capture complex interaction mechanisms and to perform prediction and classification.

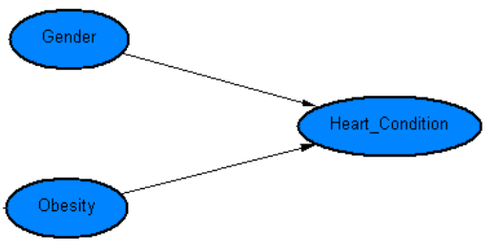
MAIN THRUST

A Bayesian network is a graph, where nodes represent stochastic variables and (arrowhead) arcs represent dependencies among these variables. In the simplest case, variables are discrete, and each variable can take a finite set of values.

Representation

Suppose we want to represent the variable *gender*. The variable gender may take two possible values: male and female. The assignment of a value to a variable is called the *state of the variable*. So, the variable gender has two states: Gender = Male and Gender = Female. The graphical structure of a Bayesian network looks like this:

Figure 1.



The network represents the notion that obesity and gender affect the heart condition of a patient. The variable obesity can take three values: yes, borderline and no. The variable heart condition has two states: true and false. In this representation, the node heart condition is said to be a *child* of the nodes gender and obesity, which, in turn, are the *parents* of heart condition.

The variables used in a Bayesian networks are stochastic, meaning that the assignment of a value to a variable is represented by a probability distribution. For instance, if we do not know for sure the gender of a patient, we may want to encode the information so that we have better chances of having a female patient rather than a male one. This guess, for instance, could be based on statistical considerations of a particular population, but this may not be our unique source of information. So, for the sake of this example, let's say that there is an 80% chance of being female and a 20% chance of being male. Similarly, we can encode that the incidence of obesity is 10%, and 20% are borderline cases. The following set of distributions tries to encode the fact that obesity increases the cardiac risk of a patient, but this effect is more significant in men than women:

The dependency is modeled by a set of probability distributions, one for each combination of states of the variables gender and obesity, called the parent variables of heart condition.

Figure 2.

Heart_Condition

Obesity	Gender	True	False
Yes	Male	0.800	0.200
Yes	Female	0.700	0.300
Borderline	Male	0.750	0.250
Borderline	Female	0.600	0.400
No	Male	0.200	0.800
No	Female	0.100	0.900

Learning

Learning a Bayesian network from data consists of the induction of its two different components: (1) the graphical structure of conditional dependencies (model selection) and (2) the conditional distributions quantifying the dependency structure (parameter estimation).

There are two main approaches to learning Bayesian networks from data. The first approach, known as constraint-based approach, is based on conditional independence tests. As the network encodes assumptions of conditional independence, along this approach we need to identify conditional independence constraints in the data by testing and then encoding them into a Bayesian network (Glymour, 1987; Pearl, 1988; Whittaker, 1990).

The second approach is Bayesian (Cooper & Herskovitz, 1992; Heckerman et al., 1995) and regards model selection as an hypothesis testing problem. In this approach, we suppose to have a set M= $\{M_0, M_1, ..., M_g\}$ of Bayesian networks for the random variables $Y_1, ..., Y_v$,, and each Bayesian network represents an hypothesis on the dependency structure relating these variables. Then, we choose one Bayesian network after observing a sample of data D = $\{y_{1k}, ..., y_{vk}\}$, for $k = 1, \ldots, n$. If $p(M_h)$ is the prior probability of model M_h, a Bayesian solution to the model selection problem consists of choosing the network with maximum posterior probability:

$$p(M_h|D) \propto p(M_h)p(D|M_h).$$

The quantity $p(M_h|D)$ is the marginal likelihood, and its computation requires the specification of a parameterization of each model M_h and the elicitation of a prior distribution for model parameters. When all variables are discrete or all variables are continuous, follow Gaussian distributions, and the dependencies are linear and the marginal likelihood factorizes into the product of marginal likelihoods of each node and its parents. An important property of this likelihood modularity is that in the comparison of models that differ only for the parent structure of a variable Y_i, only the local marginal likelihood matters. Thus, the comparison of two local network structures that specify different parents for Y_i can be done simply by evaluating the product of the local Bayes factor $BF_{h,k} = p(D|M_{hi}) / p(D|M_{ki})$, and the ratio $p(M_{hi}) / p(M_{ki})$, to compute the posterior odds of one model vs. the other as $p(M_{hi}|D) / p(M_{ki}|D)$.

In this way, we can learn a model locally by maximizing the marginal likelihood node by node. Still, the space of the possible sets of parents for each variable grows exponentially with the number of parents involved, but successful heuristic search procedures (both deterministic and stochastic) exist to render the task more amenable (Cooper & Herskovitz, 1992; Singh & Larranaga, 1996; Valtorta, 1995).