# Graphical Modeling

*Prince Afriyie and Munni Begum*

**Prince Afriyie** is currently studying for his Ph.D. in the Department of Statistics, Fox School of Business, Temple University, Philadelphia. This article was written as part of a project under the direction of Dr. Begum

**Munni Begum** is Associate Professor, Department of Mathematical Sciences, Ball State University.

**Abstract**: Graphical models have been an area of active research since the beginning of the twentieth century. Graphical models have wide scope of applicability in various scientific fields. This paper presents applications of graphical models with a focus on Bayesian networks. An exploration on the basics of graph theory and probability theory which tie together to form graphical models is outlined. Markov properties, graph decompositions, and their implications to inference are discussed. An algorithmic software for graphical models, Netica is used to demonstrate an inference problem in medical diagnostics. We address instances where parameters in the model are unknown, through maximum likelihood method if analytically feasible, but numerical and Markov Chain Monte Carlo methods are warranted otherwise.

## Introduction

A graphical model is a probabilistic model with an underlying graph denoting the conditional independence structure among stochastic components. In other words, a graphical model is a marriage between probability distribution theory and graph theory providing a natural tool for dealing with a large class of problems containing uncertainty and complexity. A complex model is built by combining simpler parts, an idea known as modularity. Graphical models are used in probability theory, statistics (Bayesian statistics, in par-

ticular), and machine learning [3]. When used in conjunction with statistical techniques, graphical models have several advantages for data analysis, since they encode dependencies among all variables and readily handle situations where some data entries are missing. A Bayesian network is a graphical model that encodes probabilistic relationships among variables of interest. Bayesian networks can be used to learn causal relationships, and hence can be used to gain understanding about a problem domain and to predict the consequences of intervention. Since Bayesian networks have both a causal and probabilistic semantics, these are ideal representations for combining prior knowledge (which often comes in causal form and expert's opinion) and data.

## Graph Terminologies

*Vertices, points or nodes* are the interconnected objects in a graph. *Edges, links, lines or arcs* are the links that connect pairs of vertices. *A graph* is a pair $G = (V, E)$, where $V$ is a (finite) set of vertices or nodes and $E \subseteq V \times V - \Delta$ is a (finite) set of edges, links, or arcs. Here $\Delta := \{(A, A) : A \in V\}$. $G$ is called *undirected* if and only if

$$\forall A, B \in V : (A, B) \in E \Longrightarrow (B, A) \in E.$$

$G$ is called *directed* if and only if

$$\forall A, B \in V : (A, B) \in E \Longrightarrow (B, A) \notin E.$$

Let $G = (V, E)$ be an undirected graph. $A$ node $B \in V$ is called *adjacent* to a node $A \in V$ or a *neighbor* of $A$ if and only if there is an edge between them, i.e., if and only if $(A, B) \in E$. The set of all neighbors of $A$ is

$$\text{neighbors}(A) = \{B \in V \mid (A, B) \in E\},$$

and $\deg(A) = |\text{neighbors}(A)|$ is the *degree* of the node $A$ (number of incident edges). The set neighbors$(A)$ is also known as the *boundary* of $A$. The boundary of $A$ together with $A$ is called the *closure* of $A$. Thus

$$\text{closure}(A) = \text{neighbors}(A) \cup \{A\}.$$

Fig. 1 is an example of an undirected graph. The edges between the nodes in Fig. 1 are all undirected.

Let $G = (V, E)$ be an undirected graph. Two distinct nodes $A, B \in V$ are called *connected* in $G$, written $A \sim B$, if and only if there exists a sequence $C_1, C_2, ..., C_k, k \geq 2$, of distinct nodes, called a *path*, with $C_1 = A, C_k = B$, and $\forall i, 1 \leq i \leq k \mid (C_i, C_{i+1}) \in E$. An undirected graph is called *singly connected* or a *tree* if and only if any pair of distinct nodes is connected by exactly one path. Let $G = (V, E)$ be an undirected graph. An undirected graph $G_X = (X, E_X)$ is called a *subgraph* of $G$ (induced by $X$) if and only if $X \subseteq V$ and $E_X = (X \times X) \cap E$, that is, if and only if it contains a subset of the nodes in $G$ and all corresponding edges. An undirected graph $G = (V, E)$ is called complete if and only if its set of edges is complete, that is, if and only if all possible edges are present, or

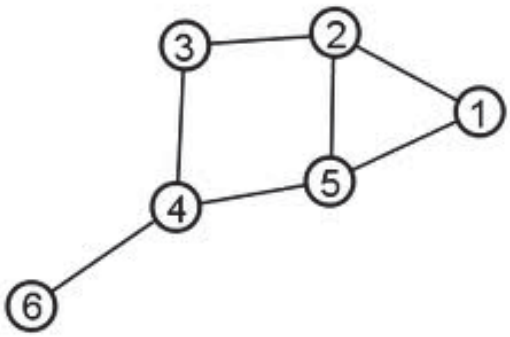


Figure 1: An undirected graph

formally if and only if, $E = V \times V - \{(A, A) \mid A \in V\}$. A complete subgraph is called a *clique*. A clique is called *maximal* if and only if it is not a subgraph of a larger clique, that is, a clique having more nodes.

Let $G = (V, E)$ be a directed graph. A node $B \in V$ is called a *parent* of a node $A \in V$ and, conversely, $A$ is called the *child* of $B$ if and only if there is a directed edge from $B$ to $A$, that is, if and only if $(B, A) \in E$. The set of all *parents* of a node A is denoted by

$$\text{parents}(A) = \{B \in V \mid (B, A) \in E\},$$

and the set of its *children* is denoted

$$\text{children}(A) = \{B \in V \mid (A, B) \in E\}.$$

$B$ is called *adjacent to* $A$ if and only if it is either a parent or a child of $A$. A *directed acyclic graph* (commonly abbreviated to DAG), is a directed graph with no directed cycles. Notice in Fig. 2 that the direction of the edges does not follow a cycle, hence Fig. 2 is an example of a DAG.

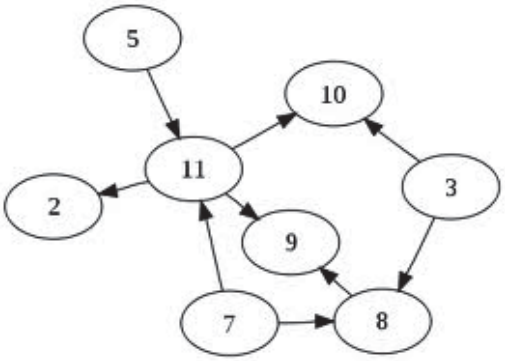


Figure 2: A directed acyclic graph

Let $G = (V, E)$ be a directed acyclic graph. A node $A \in V$ is called an *ancestor* of another node $B \in V$ and, conversely, $B$ is called a *descendant* of $A$ if and only if there is a directed path from $A$ to $B$. The set of all ancestors of a node $A$ is denoted by

$$\text{ancestors}(A) = \{B \in V \mid B \sim A\},$$

the set of its descendants is denoted by

$$\text{descendants}(A) = \{B \in V \mid A \sim B\}.$$

$B$ is called a *non-descendant* of $A$ if and only if it is distinct from $A$ and not a descendant of $A$. The set of its non-descendants is denoted by

$$\text{nondescs}(A) = V - \{A\} - \text{descendants}(A).$$

A directed acyclic graph is called *singly connected* or a *polytree* if and only if each pair of distinct nodes is connected by exactly one path. A directed acyclic graph is called a *(directed) tree* if and only if it is a polytree and exactly one node (the so-called *root node*) has no parents.

Let $G = (V, E)$ be a directed acyclic graph. A numbering of the nodes of $G$, that is, a function $o : V \to \{1, ..., |V|\}$ satisfying

$$\forall A, B \in V \; (A, B) \in E \Rightarrow o(A) \leq o(B)$$

is called a *topological order* of the nodes of $G$. Let $G = (V, E)$ be a directed acyclic graph and $X, Y$, and $Z$ three disjoint subsets of nodes. $Z$ *d-separates* $X$ and $Y$ in $G$, if and only if there is no path from a node in $X$ to a node in $Y$ along which the following two conditions hold: (1) every node with converging edges (from its predecessor and its successor on the path) either is in $Z$ or has a descendant in $Z$, (2) every other node is not in $Z$. A path satisfying the conditions above is said to be *active*, otherwise it is said to be *blocked* (by $Z$); so separation means that all paths are blocked. Let $G = (V, E)$ be an undirected graph and $X, Y$, and $Z$ three disjoint subsets of nodes (vertices). $Z$ *u-separates* $X$ and $Y$ in $G$, written $\langle X \mid Z \mid Y \rangle$, iff all paths from a node in $X$ to a node in $Y$ contain a node in $Z$. A path that contains a node in $Z$ is called *blocked* (by $Z$), otherwise it is called *active*; so separation means that all paths are blocked [1].

## Markov Properties of Graphical Models

### Markov properties of undirected graphs

Let $(\cdot \perp\!\!\!\perp_\delta \cdot \mid \cdot)$ be a three-place relation representing the set of conditional independence statements that hold in a given joint distribution $\delta$ over a set $U$ of attributes. An undirected graph $G = (U, E)$ is said to have (with respect to the distribution $\delta$) the:

(i) *pairwise Markov property* if and only if in $\delta$ any pair of attributes, which are nonadjacent in the graph, are conditionally independent given all remaining attributes, that is, if and only if

$$\forall A, B \in U,\ A \neq B : (A, B) \notin E \Rightarrow A \perp\!\!\!\perp_\delta B \mid U - \{A, B\};$$

(ii) *local Markov property* if and only if in $\delta$ any attribute is conditionally independent of all remaining attributes given its neighbors, that is, if and only if

$$\forall A \in U : A \perp\!\!\!\perp_\delta U - \text{closure}(A) \mid \text{neighbors}(A);$$

(iii) *global Markov property* if and only if in $\delta$ any two sets of attributes which are u-separated by a third are conditionally independent given the attributes in the third set, that is, if and only if

$$\forall X, Y, Z \subseteq U : \langle X \mid Z \mid Y \rangle \Rightarrow X \perp\!\!\!\perp_\delta Y \mid Z.$$

**Markov properties of directed graphs**

We define the Markov properties of a directed graph along the same line. Let $(\cdot \perp\!\!\!\perp_\delta \cdot \mid \cdot)$ be a three-place relation representing the set of conditional independence statements that hold in a given joint distribution $\delta$ over a set $U$ of attributes. A directed acyclic graph $G = (U, E)$ is said to have (with respect to the distribution $\delta$) the:

(i) *pairwise Markov property* if and only if in $\delta$ any attribute is conditionally independent of any non-descendant not among its parents given all remaining non-descendants, that is, if and only if

$$\forall A, B \in U : B \in \text{nondescs}(A) - \text{parents}(A) \Rightarrow A \perp\!\!\!\perp_\delta B \mid \text{nondescs}(A) - \{B\};$$

(ii) *local Markov property* if and only if in $\delta$ any attribute is conditionally independent of all remaining non-descendants given its parents, that is, if and only if

$$\forall A \in U : A \perp\!\!\!\perp_\delta \text{nondescs}(A) - \text{parents}(A) \mid \text{parents}(A);$$

(iii) *global Markov property* if and only if in $\delta$ any two sets of attributes which are d-separated by a third are conditionally independent given the attributes in the third set, that is, if and only if

$$\forall X, Y, Z \subseteq U : \langle X \mid Z \mid Y \rangle \Rightarrow X \perp\!\!\!\perp_\delta Y \mid Z.$$

**Graphs and Decompositions** A probability distribution $p_U$ over a set $U$ of attributes is called decomposable or factorizable with respect to a directed acyclic graph $G = (U, E)$ if and only if it can be written as a product of the

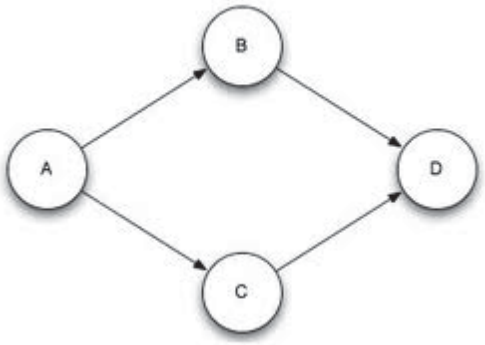


Figure 3: Graph decomposition

conditional probabilities of the attributes given their parents in $G$ [1]. For example, the graph

corresponds to the factorization

$$Pr(A, B, C, D) = Pr(A) * Pr(B|A) * Pr(C|A) * Pr(D|B, C).$$

Decomposability of a graph has important implications in computational statistical inference since the joint probability distribution of attributes can be factored into simpler conditional and marginal probability distributions and optimization can be carried out.

## Bayesian networks

A Bayesian network is a directed conditional independence graph of a probability distribution together with the family of conditional probabilities from factorization induced by a directed acyclic graph. Bayesian networks explicitly depict uncertainty as probabilities, which can fit well in a risk analysis and risk management framework. Bayesian networks can be used to help identify key factors that influence some outcome of interest, to help prioritize monitoring or research. A Bayesian network characterizes the joint probability distribution of a set of random variables using conditional independence relationships among them [5]. Bayesian networks are based on directed acyclic graphs along with a set of conditional probability tables representing conditional independence relationships for each node in these graphs. The conditional probabilities for each child is computed based on Markov condition or the local Markov property for directed graphs discussed in section 2. Computation of probabilities using a Bayesian network can be referred to as *inference*. In general, inference involves queries of the form: $Pr(X \mid E)$ where $X$ is the query variable and $E$ is the evidence variable.

## Estimating Parameters in a Bayesian network

The aim of learning is to predict the nature of future data based on past experience. One can construct a probabilistic model for a situation where the model contains unknown parameters. The 'classical' approach of parameter estimation regards a parameter as fixed. In this case, a parameter is unknown and has to be estimated, so it is not considered to be a random variable. One therefore computes approximations to the unknown parameters, and uses these to compute an approximation to the probability density. The parameter is considered to be fixed and unknown, because there is usually a basic assumption that in ideal circumstances, the experiment could be repeated infinitely many times and the estimating procedure would return a precise value for the parameter. That is, if one increases the number of replications indefinitely, the estimate of the unknown parameter converges, with probability one, to the true value. This is known as the 'frequentist' interpretation of probability.

The Bayesian approach takes the view that since the parameter is unknown, it is a random variable. A probability distribution, known as the prior distribution, is put over the parameter space, based on a prior assessment of where the parameter may lie. One then carries out an experiment and using the available data, one applies Bayes rule to compute posterior distribution, which is the updated probability distribution over the parameter space. The posterior distribution is obtained as: $\textit{posterior} \propto \textit{Likelihood} \times \textit{prior}$ and is then used to compute the probability distribution for future events, based on past experience. Unlike the classical approach, this is an exact distribution, but it contains a subjective element which is described by the prior distribution [1]. Characteristics of a posterior distribution such as posterior mode and median are used as parameter estimates with credible sets as frequentist version of confidence limits. Analytical methods based on conjugate priors in simpler case, and sampling based iterative Markov Chain Monte Carlo methods in non-tractable cases are used to carry out such computation.

In Bayesian statistics, computation of posterior distribution usually requires iterative numerical methods and Markov chain Monte Carlo methods. These are similar to 'frequentist' approach in the sense that they rely upon an arbitrarily large supply of independent random numbers to obtain the desired precision. From an engineering point of view, there are efficient pseudo-random number generators that supply arbitrarily large sequences of 'random' numbers of very good quality. That is, there are tests available to show whether a sequence 'behaves' like an observation of a sequence of suitable independent random numbers. Both approaches to statistical inference have an arbitrary element. For the classical approach, one sees this in the choice of sample space. According to Jeffrey[5] a sample space is the set of observations that one wants to work with, but may no be able to choose due to practical constraints. Similarly, in Bayesian approach it is not always straight forward to determine how long a random sequence should be in order to achieve convergence, while estimating the parameters of a posterior distribution.

## Applications of Bayesian networks

Exact inference is feasible in small to medium-sized networks. Exact inference in large networks takes relatively longer time. We resort to approximate inference techniques which may be faster and may produce good results. Computation involved in inferences in a Bayesian network can be performed with packages: Netica, GeNle, HUGIN, Elvira and BUGS/WinBUGS. In most of these packages, information about the observed value of a variable is propagated through the network to update the probability distributions over other variables that are not observed directly. The law of total probability is used in the 'forward' propagation case, that is computing conditional probabilities of each child while instantiating the parent. Using Bayes rule, influences may also be identified in a 'backwards' direction, from dependent variables to their parents [5].

Fig.4 illustrates an example of a Bayesian network, taken from Cowell et al [3]. In this example, dyspnea may be caused by tuberculosis, lung cancer or bronchitis, or none of them, or more than one of them. A visit to Asia increases the chances of tuberculosis, while smoking is known to be a risk factor for both lung cancer and bronchitis. The results of a single chest X-ray do not discriminate between lung cancer and tuberculosis, as neither does the presence or absence of dyspnea.

The model might be applied to the following hypothetical situation. A patient with dyspnea, who had recently been to Asia, shows up at a Chest clinic? Smoking history and chest X-ray are not yet available. The doctor would like to know the chance that each of these diseases is present, and if tuberculosis were ruled out by another test, how would that change the belief in lung cancer? Also, would knowing smoking history or getting an X-ray contribute more information about cancer, given that smoking may 'explain away' the dyspnea since bronchitis is considered a possibility? Finally, when all information is in, can we identify which was the most influential in forming our judgment [4]? All these questions and possibilities can be answered by inference (by means of algorithms) from the model. Fig. 5 illustrates Netica being used in this same example with conditional probabilities for each node [7].

## Further applications

Bayesian networks can be applied to social network analysis to derive insights that are not possible using traditional social network analysis techniques. We discuss three types of analyses that are enabled using Bayesian networks: augmenting social network algorithms with uncertainty, searching the network for nodes, and inferring new links in the network.

Traditional social network graph theoretic algorithms do not take uncertainty into account. While a node may appear to have a high value for degree centrality, the algorithm does not consider the certainty of the links, authority from whom the link information was gathered, recency of the link, or any other type of meta-information (i.e., qualifiers of the information) that may be known Carleson et al., 2006 ([2]).

Bayesian networks can augment social network algorithms by considering meta-information in their calculations. For example, the user of a social network tool that incorporates uncertainty might be interested in determining the 'importance' of each individual in the network. The user would create a Bayesian network for 'importance', which might contain one node representing the algorithmic degree centrality computation, and another node that represents the total certainty of the data used in the calculation. These two nodes might be parents of the 'importance' node, which the user would provide with a set of conditional probability entries. In addition, due to the abductive reasoning capabilities of Bayesian networks, one could investigate questions such as, 'What might be required for this individual to increase in importance?' by setting the value on an individual's 'importance' node to a value, and observing what values the parent nodes would need to support that belief.

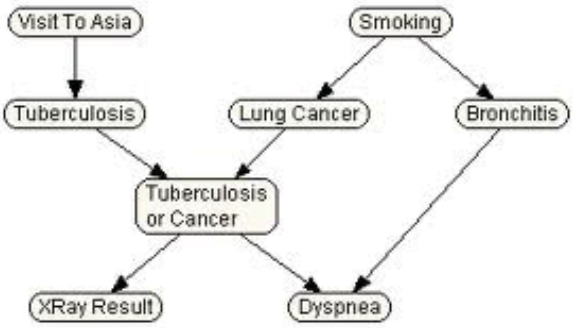


Figure 4: A Bayesian network for medical diagnosis

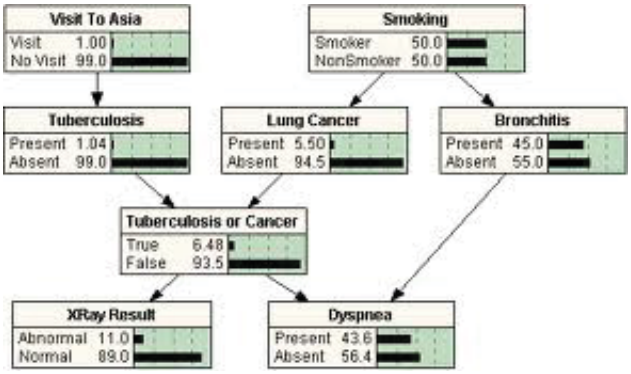


Figure 5: Netica inference engine for medical diagnosis problem

Bayesian networks can be applied to all individuals in a social network. The user can find and sort results of individuals of interest in a social network. This is particularly useful when the user is working with a large network (e.g.,

email traffic in a multinational corporation), and wants to find nodes that fit a particular set of attributes. For example, a user might be interested in individuals within the network that are likely to become future leaders in the organization. This is different from searching for simple node attributes, such as 'Name' or 'Age', because the notion of 'Leadership Potential' is a psychosocial concept based on a combination of other attributes and relationships that cannot be handled by a simple search capability. Some of those attributes or relationships may be associated with a degree of uncertainty [6]. Prominent examples of social networks today include *facebook, myspace*, and *hi*5.

### Bayesian Networks with Continuous random variables

So far, we have discussed about Bayesian networks with discrete random variables. We can also have a Bayesian network with continuous random variables. An example is a Gaussian Bayesian network which contains variables that are normally distributed. In a Gaussian Bayesian network, the parents are normally distributed and each child is a linear function of its parents, plus an error term which is normally distributed with mean zero and variance $\sigma^2$. For instance if $x_1, x_2, ..., x_n$ are the parents of $Y$, then

$Y = b_1x_1 + b_2x_2 + ... + b_nx_n + \epsilon$, where $\epsilon \sim N(0, \sigma^2)$

$Y$ is distributed conditionally as

$y \mid x \sim N(b_1x_1 + b_2x_2 + ... + b_nx_n, \sigma^2)$.

There are exact inference algorithm for Gaussian Bayesian networks. Most Bayesian network inference algorithms like Netica and HUGIN handle Gaussian Bayesian networks. HUGIN uses the exact algorithm while Netica discretizes the continuous distribution and then does inference using discrete variables [7].

## Conclusion

Graphical models are probabilistic models for which a graph denotes the conditional independence structure between random variables. They provide a natural tool for dealing with two problems that occur throughout applied mathematics and engineering, uncertainty and complexity, and in particular they play an increasingly important role in the design and analysis of machine learning algorithms [5]. Using the concepts of graph theory and probability theory we can simplify complex systems to make it easier to solve practical problems. Unknown parameters in a graphical model can be estimated by the method of maximum likelihood, numerical and Markov Chain Monte Carlo methods. Graphical modeling, especially Bayesian networks, have a wide scope of applicability in vast fields due to the fact that inferences can easily be made based on the model. Areas of applicability include military, industrial, medical diagnostics, and commercial especially in computer software engineering.

## References


[1] Christian Borgelt, Matthias Steinbrecher, and Rudolf Kruse, Graphical models, methods for data analysis and mining, 2nd ed. Wiley, 2009.

[2] Eric Carlson, Sean Guarino, Jonathan Pfautz, Methods for Representing Bias in Bayesian Networks, 2006. Retrieved from `http://www.cs.uu.nl/groups/DSS/UAI08-workshop/Papers/BMAW-CGP.pdf`

[3] R. G. Cowell, A. P. Dawid, S. L. Lauritzen, and D. J. Spiegelhalter, Probabilistic Networks and Expert Systems, Springer, 1998.

[4] David Edwards, Introduction to Graphical Modeling, 2nd ed. Springer, 1949

[5] Michael I. Jordan, *Statistical Science, Graphical models*, **19** (1997) 140–155.

[6] David Koelle, Jonathan Pfautz, Michael Farry, Zach Cox, Geoffrey Catto, and Joseph Campolongo, Applications of Bayesian Belief Networks in Social Network Analysis, 2008. Retrieved from `http://www.cs.uu.nl/groups/DSS/UAI-workshop/Koelle.pdf`

[7] Richard E Neapolitan, Probabilistic methods for bioinformatics, Elsevier Inc., 2009