Bayesian Networks for Genomic Analysis

Paola Sebastiani* Maria M. Abad† Marco F. Ramoni‡

November 2, 2004

Abstract

Bayesian networks are emerging into the genomic arena as a general modeling tool able to unravel the cellular mechanism, to identify genotypes that confer susceptibility to disease, and to lead to diagnostic models. This chapter reviews the foundations of Bayesian networks and shows their application to the analysis of various types of genomic data, from genomic markers to gene expression data. The examples will highlight the potential of this methodology but also the current limitations and we will describe new research directions that hold the promise to make Bayesian networks a fundamental tool for genome data analysis.

*Department of Biostatistics, Boston University School of Public Health, 715 Albany Street, Boston MA 02118. Email: sebas@bu.edu

†Software Engineering Department, University of Granada, Daniel Saucedo Aranda, Granada, 18071 Spain. Email: mabad@ugr.es

‡Children's Hospital Informatics Program and Harvard Partners Center for Genetics and Genomics, Harvard Medical School, HMS New Research Building, 77 Pasteur Avenue, Suite 255, Boston, MA 02115. Email: marco_ramoni@harvard.edu

Contents

1 Introduction

One of the most striking characteristics of today's biomedical research practice is the availability of genomic-scale information. This situation has been created by the simultaneous but not unrelated development of "genome-wide" technologies, mostly rooted in the Human Genome Project: fast sequencing techniques, high-density genotype maps, DNA and protein microarrays. Sequencing and genotyping techniques have evolved into powerful tools to identify genetic variations across individuals responsible for predispositions to some disease, response to therapies, and other observable characters known as phenotypes. Single-nucleotide polymorphisms (SNPs) —a single base variation across the individuals of a population— are considered the most promising natural device to uncover the genetic basis of common diseases. By providing a high-resolution map of the genome they allow researchers to associate variations in a particular genomic region to observable traits [15, 52]. Commercially available technology, such as the Affymetrix GeneChip Mapping 10K Array and Assay Set (http://affymetrix.com), is able to simultaneously genotype 10,000 SNPs in an individual. Other technologies are able to interrogate the genomic structure of a cell on a genome-wide scale: CGH microarrays are able to provide genome-wide identification of chromosomal imbalances —such as deletions and amplifications— that are common rearrangements in most tumors [6]. These rearrangements identify different tumor types or stages and this technology allows us to dive into the mutagenic structure of tumor tissues.

Despite their differences —large scale genotyping interrogates the normal DNA of an individual, while CGH microarrays are specifically designed to study mutagenic tissues like tumors— these two technologies focus on the identification of structural genomic information, that is, information about the DNA sequence of a cell. The functional counterparts of these genomic platforms, on the other hand, are designed to quantify the expression of the genes encoded by the DNA of a cell, as amount of RNA produced by each single gene. cDNA and oligonucleotide microarrays [61, 81, 83] enable investigators to simultaneously measure the expression of thousands of genes and hold the promise to cast new light onto the regulatory mechanisms of the genome [51]. The ability they offer to observe the genome in action has opened the possibility of profiling gene behaviors, studying interactions among genes, and discovering new classes of diseases on the basis of their genomic profile alone. The rising field of proteomics takes this study one step forward to proteins —the final product of gene expression [71]— and, using mass spectrometry technology, investigators can now measure in parallel the entire protein complement in a given cell, tissue or organism [1].

All these technologies come to join, today, long-term cohort studies, like the Nurses' Health Study (http://www.channing.harvard.edu/nhs) and the Framingham Heart Study (http://www.framingham.com/heart) that have been collecting detailed "phenome-wide" information about hundred of thousands individuals over several decades. Although the individual contribution of each technology has been already invaluable, the potential of their integration is even greater, but their ability to deliver on their promise of understanding the fundamental rules of life and diseases rests on our ability to integrate this genomic information with large-scale phenotypic data [15]. The integration of information about genotypes, RNA expression, proteins and phenotypes into a coherent landscape will lead not only to the discovery of clinical phenomena not observable at each individual level but also to a better understanding of the coding and regulatory mechanisms underpinning the expression of genes [27].

The main challenge of this endeavor is the identification of a common formalism able to model this massive amount of data. Bayesian networks (also known as directed graphical models) are a knowledge representation formalism born at the confluence of artificial intelligence and statistics that offer a powerful framework to model these different data sources. Bayesian networks have been already applied, by us and others, to the analysis of different types of genomic data —from gene expression microarrays [28, 31, 70, 92, 93] to protein-protein interactions [46] and genotype data [7, 90] — and their modular nature makes them easily extensible to the task of modeling these different types of data. However, the application of Bayesian networks to genomics requires the methodological development of new statistical and computational capabilities able to capture the complexity of genomic information.

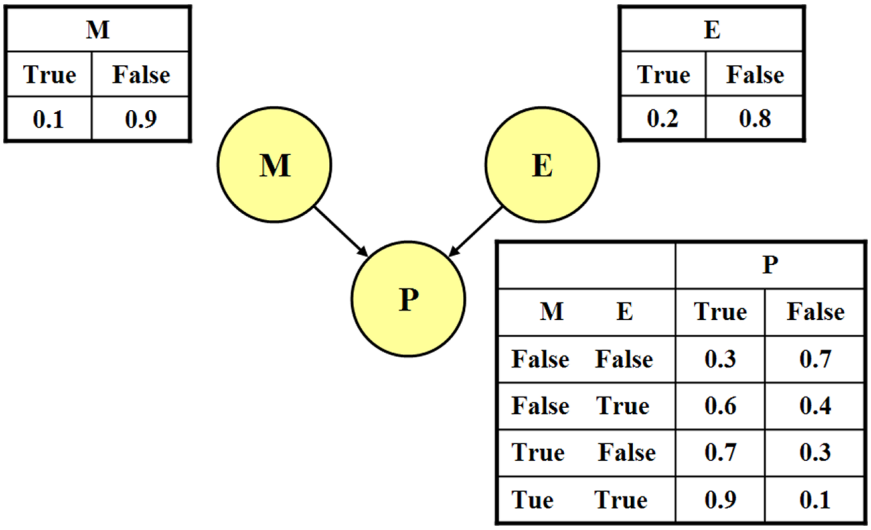


Figure 1: *A network describing the impact of a genetic marker (node M) and an environmental factor (node E) on a phenotypic character (node P). Each node in the network is associated with a probability table that describes the conditional distribution of the node, given its parents.*

This chapter will first describe the current state of the art about learning Bayesian networks from data. We will show the potential benefit of Bayesian networks as model and reasoning tool through several examples. The examples will also highlight the limitations of the current methodology and we will describe new research directions that hold the promise to make Bayesian networks a fundamental tool for genomic data analysis.

2 Fundamentals of Bayesian Networks

Bayesian networks are a representation formalism at the cutting edge of knowledge discovery and data mining [43, 63, 64]. In this section, we will review the formalism of Bayesian networks and the process of learning them from databases.

2.1 Representation and Reasoning

A Bayesian network has two components: a directed acyclic graph and a probability distribution. Nodes in the directed acyclic graph represent stochastic variables and arcs represent directed dependencies among variables that are quantified by conditional probability distributions.

As an example, consider the simple scenario in which a genetic marker together with an environmental condition create a phenotypic character. We describe the marker in the genetic code, the environmental condition, and the phenotypic character with three variables M, E, and P, each having two states "True" and "False". The Bayesian network in Figure 1 describes the dependency of the three variables with a directed acyclic graph, in which the two arcs pointing to the node P represent the joint action of the genetic marker and the environmental condition. Also, the absence of any directed arc between the genetic marker and the environmental condition describes the *marginal independence* of the two variables that become dependent when we condition on the phenotype. Following the direction of the arrows, we call the node P a *child* of M and E, which become its *parents*. The Bayesian network in Figure 1 let us decompose the overall joint probability distribution of the three variables that would consist of $2^3-1 = 7$ parameters into three probability

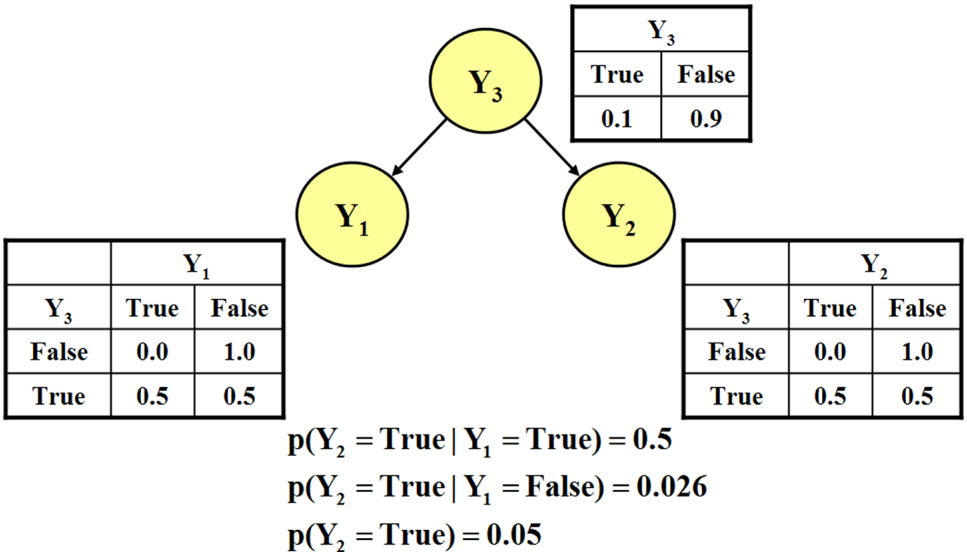


Figure 2: *A network encoding the conditional independence of Y_1, Y_2 given the common parent Y_3. The panel in the middle shows that the distribution of Y_2 changes with Y_1 and hence the two variables are conditionally dependent.*

distributions, one conditional distribution for the variable P given the parents, and two marginal distributions for the two parent variables M and E. These probabilities are specified by $1 + 1 + 4 = 6$ parameters. The decomposition is one of the key factors to provide both a verbal and a human understandable description of the system and to efficiently store and handle this distribution, which grows exponentially with the number of variables in the domain. The second key factor is the use of *conditional independence* between the network variables to break down their overall distribution into connected modules.

Suppose we have three random variables Y_1, Y_2, Y_3. Then Y_1 and Y_2 are independent given Y_3 if the conditional distribution of Y_1, given Y_2, Y_3 is only a function of Y_3. Formally:

$$p(y_1|y_2, y_3) = p(y_1|y_3)$$

where $p(y|x)$ denotes the conditional probability/density of Y, given $X = x$. We use capital letters to denote random variables, and small letters to denote their values. We also use the notation $Y_1 \perp Y_2|Y_3$ to denote the conditional independence of Y_1 and Y_2 given Y_3.

Conditional and marginal independence are substantially different concepts. For example two variables can be marginally independent, but they may be dependent when we condition on a third variable. The directed acyclic graph in Figure 1 shows this property: the two parent variables are marginally independent, but they become dependent when we condition on their common child. A well known consequence of this fact is the Simpson's paradox [105] and a typical application in genetics is the dependency structure of genotypes among members of the same family: the genotypes of two parents are independent, assuming random mating, but they become dependent once the genotype of their common child is known.

Conversely, two variables that are marginally dependent may be made conditionally independent by introducing a third variable. This situation is represented by the directed acyclic graph in Figure 2, which shows two children nodes (Y_1 and Y_2) with a common parent Y_3. In this case, the two children nodes are independent, given the common parent, but they may become dependent when we marginalize the common parent out. Suppose, for example, the three variables represent the presence/absence of an X-linked genetic

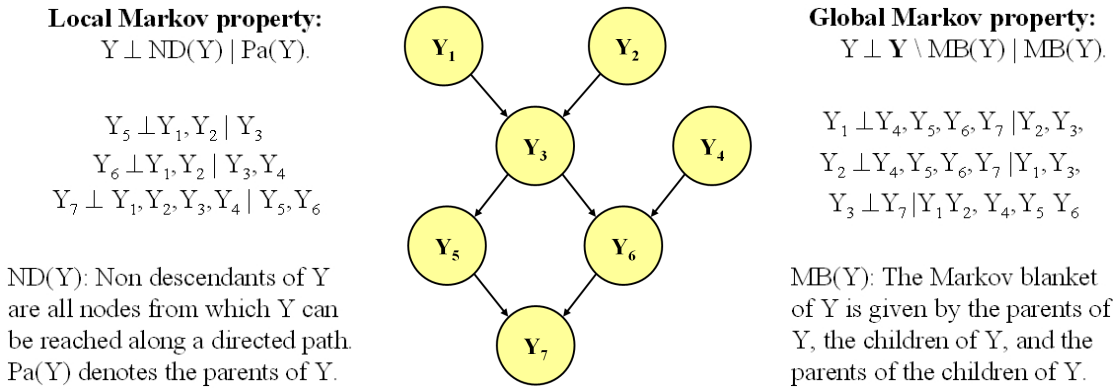


Figure 3: *A Bayesian network with seven variables and some of the Markov properties represented by its directed acyclic graph. The panel on the left describes the local Markov property encoded by a directed acyclic graph and lists the three Markov properties that are represented by the graph in the middle. The panel on the right describes the global Markov property and lists three of the seven global Markov properties represented by the graph in the middle. The vector in bold denotes the set of variables represented by the nodes in the graph.*

marker in the mother genotype (Y_3) and the children genotype (Y_1 and Y_2). The marginal distribution of Y_3 represents the prevalence of the marker in the population, and the conditional probabilities associated with the nodes Y_1 and Y_2 represent the probability that each child has the marker, given the maternal genotype. Then it is easy to compute the conditional probability that one of the two children has the marker, given that only the genotype of the other child is known. Because the probability of Y_2 changes according to the value of Y_1, the two variables are dependent. The seminal papers by Dawid [19, 20] summarize many important properties and alternative definitions of conditional independence.

The overall list of marginal and conditional independencies represented by the directed acyclic graph is summarized by the local and global Markov properties [57] that are exemplified in Figure 3 using a network of seven variables. The *local Markov property* states that each node is independent of its non descendant given the parent nodes and leads to a direct factorization of the joint distribution of the network variables into the product of the conditional distribution of each variable Y_i given its parents $Pa(y_i)$. Therefore, the joint probability (or density) of the v network variables can be written as:

$$p(y_1, ..., y_v) = \prod_i p(y_i|pa(y_i)). \tag{1}$$

In this equation, $pa(y_i)$ denotes a set of values of $Pa(Y_i)$. This property is the core of many search algorithms for learning Bayesian networks from data. With this decomposition, the overall distribution is broken into modules that can be interrelated, and the network summarizes all significant dependencies without information disintegration. Suppose, for example, the variable in the network in Figure 3 are all categorical. Then the joint probability $p(y_1, \ldots, y_7)$ can be written as the product of seven conditional distributions:

$$p(y_1)p(y_2)p(y_3|y_1, y_2)p(y_4)p(y_5|y_3)p(y_6|y_3, y_4)p(y_7|y_5, y_6).$$

The *global Markov property*, on the other hand, summarizes all conditional independencies embedded by the directed acyclic graph by identifying the Markov Blanket of each node. This property is the foundation of many algorithms for probabilistic reasoning with Bayesian networks that allow the investigation of undirected relationships between the variables, and their use for making prediction and explanation. In the network

in Figure 3, for example, we can compute the probability distribution of the variable Y_7, given that the variable Y_1 is observed to take a particular value (prediction) or, vice versa, we can compute the conditional distribution of Y_1 given the values of some other variables in the network (explanation). In this way, a Bayesian network becomes a complete simulation system able to forecast the value of unobserved variables under hypothetical conditions and, conversely, able to find the most probable set of initial conditions leading to observed situation. Exact algorithms exist to perform this inference when the network variables are all discrete, all continuous and modelled with Gaussian distributions, or the network topology is constrained to particular structures [9, 58, 69].

For general network topologies and non standard distributions, we need to resort to stochastic simulation [12]. Among the several stochastic simulation methods currently available, Gibbs sampling [36, 103] is particularly appropriate for Bayesian network reasoning because of its ability to leverage on the graphical decomposition of joint multivariate distributions to improve computational efficiency. Gibbs sampling is also useful for probabilistic reasoning in Gaussian networks, as it avoids computations with joint multivariate distributions. Gibbs sampling is a Markov Chain Monte Carlo method that generates a sample from the joint distribution of the nodes in the network. The procedure works by generating an ergodic Markov chain

$$\begin{pmatrix} y_{10} \\ \vdots \\ y_{v0} \end{pmatrix} \rightarrow \begin{pmatrix} y_{11} \\ \vdots \\ y_{v1} \end{pmatrix} \rightarrow \begin{pmatrix} y_{12} \\ \vdots \\ y_{v2} \end{pmatrix} \rightarrow \cdots$$

that, under regularity conditions, converges to a stationary distribution. At each step of the chain, the algorithm generates y_{ik} from the conditional distribution of Y_i given all current values of the other nodes. To derive the marginal distribution of each node, the initial burns-in is removed, and the values simulated for each node are a sample generated from the marginal distribution. When one or more nodes in the networks are observed, they are fixed in the simulation so that the sample for each node is from the conditional distribution of the node given the observed nodes in the network.

Gibbs sampling in directed graphical models exploits the Global Markov property, so that to simulate from the conditional distribution of one node Y_i given the current values of the other nodes, the algorithm needs to simulate from the conditional probability/density

$$p(y_i|y \backslash y_i) \propto p(y_i|pa(y_i)) \prod_h p(c(y_i)_h|pa(c(y_i)_h))$$

where y denotes a set of values of all network variables, $pa(y_i)$ and $c(y_i)$ are values of the parents and children of Y_i, $pa(c(y_i)_h)$ are values of the parents of the hth child of Y_i, and the symbol $\backslash$ denotes the set difference.

2.2 Learning Bayesian Networks from Data

Learning a Bayesian network from data consists of the induction of its two different components: 1) The graphical structure of conditional dependencies (*model selection*); 2) The conditional distributions quantifying the dependency structure (*parameter estimation*). While the process of parameter estimation follows quite standard statistical techniques (see [73]), the automatic identification of the graphical model best fitting the data is a more challenging task. This automatic identification process requires two components: a scoring metric to select the best model and a search strategy to explore the space of possible, alternative models. This section will describe these two components — model selection and model search — and will also outline some methods to validate a graphical model once it has been induced from a data set.

2.2.1 Scoring Metrics

We describe the traditional Bayesian approach to model selection that solves the problem as hypothesis testing. Other approaches based on independence tests or variants of the Bayesian metric like the minimum description length (MDL) score or the Bayesian information criterion (BIC) are described in [57, 98, 105]. We suppose to have a set $\mathcal{M} = \{M_0, M_1, ..., M_g\}$ of Bayesian networks, each network describing an hypothesis on the dependency structure of the random variables $Y_1, ..., Y_v$. Our task is to choose one network after observing a sample of data $\mathcal{D} = \{y_{1k}, ..., y_{vk}\}$, for $k = 1, \ldots, n$. By Bayes' theorem, the data $\mathcal{D}$ are used to revise the prior probability $p(M_h)$ of each model into the posterior probability, which is calculated as

$$p(M_h|\mathcal{D}) \propto p(M_h)p(\mathcal{D}|M_h)$$

and the Bayesian solution consists of choosing the network with maximum posterior probability. The quantity $p(\mathcal{D}|M_h)$ is called the *marginal likelihood* and is computed by averaging out θ_h from the likelihood function $p(\mathcal{D}|\theta_h)$, where Θ_h is the vector parameterizing the distribution of $Y_1, ..., Y_v$, conditional on M_h. Note that, in a Bayesian setting, Θ_h is regarded as a random vector, with a prior density $p(\theta_h)$ that encodes any prior knowledge about the parameters of the model M_h. The likelihood function, on the other hand, encodes the knowledge about the mechanism underlying the data generation. In our framework, the data generation mechanism is represented by a network of dependencies and the parameters are usually a measure of the strength of these dependencies. By averaging out the parameters, the marginal likelihood provides an overall measure of the data generation mechanism that is independent of the values of the parameters. Formally, the marginal likelihood is the solution of the integral

$$p(\mathcal{D}|M_h) = \int p(\mathcal{D}|\theta_h)p(\theta_h)d\theta_h.$$

The computation of the marginal likelihood requires the specification of a parameterization of each model M_h that is used to compute the likelihood function $p(\mathcal{D}|\theta_h)$, and the elicitation of a prior distribution for Θ_h. The local Markov properties encoded by the network M_h imply that the joint density/probability of a case k in the data set can be written as

$$p(y_{1k}, ..., y_{vk}|\theta_h) = \prod_i p(y_{ik}|pa(y_i)_k, \theta_h). \tag{2}$$

Here, $y_{1k}, ..., y_{vk}$ is the set of values (*configuration*) of the variables for the kth case, and $pa(y_i)_k$ is the configuration of the parents of Y_i in case k. By assuming exchangeability of the data, that is, cases are independent given the model parameters, the overall likelihood is then given by the product

$$p(\mathcal{D}|\theta_h) = \prod_{ik} p(y_{ik}|pa(y_i)_k, \theta_h).$$

Computational efficiency is gained by using priors for Θ_h that obey the Directed Hyper-Markov law [21]. Under this assumption, the prior density $p(\theta_h)$ admits the same factorization of the likelihood function, namely $p(\theta_h) = \prod_i p(\theta_{hi})$, where θ_{hi} is the subset of parameters used to describe the dependency of Y_i on its parents. This parallel factorization of the likelihood function and the prior density allows us to write

$$p(\mathcal{D}|M_h) = \prod_{ik} \int p(y_{ik}|pa(y_i)_k, \theta_{hi})p(\theta_{hi})d\theta_{hi} = \prod_i p(D|M_{hi})$$

where $p(D|M_{hi}) = \prod_k \int p(y_{ik}|pa(y_i)_k, \theta_{hi})p(\theta_{hi})d\theta_{hi}$. By further assuming decomposable network prior probabilities that factorize as $p(M_h) = \prod_i p(M_{hi})$ [44], the posterior probability of a model M_h is the product:

$$p(M_h|\mathcal{D}) = \prod_i p(M_{hi}|\mathcal{D}).$$

Here $p(M_{hi}|\mathcal{D})$ is the posterior probability weighting the dependency of Y_i on the set of parents specified by the model M_h. Decomposable network prior probabilities are encoded by exploiting the modularity of a Bayesian network, and are based on the assumption that the prior probability of a local structure M_{hi} is independent of the other local dependencies M_{hj} for $j \neq i$. By setting $p(M_{hi}) = (g+1)^{-1/v}$, where $g+1$ is the cardinality of the model space and v is the cardinality of the set of variables, there follows that uniform priors are also decomposable.

An important consequence of the likelihood modularity is that, in the comparison of models that differ for the parent structure of a variable Y_i, only the local marginal likelihood matters. Therefore, the comparison of two local network structures that specify different parents for the variable Y_i can be done by simply evaluating the product of the local *Bayes factor* $BF_{hk} = p(\mathcal{D}|M_{hi})/p(\mathcal{D}|M_{ki})$, and the prior odds $p(M_h)/p(M_k)$, to compute the posterior odds of one model versus the other:

$$p(M_{hi}|\mathcal{D})/p(M_{ki}|\mathcal{D}).$$

The posterior odds provide an intuitive and widespread measure of fitness. Another important consequence of the likelihood modularity is that, when the models are a priori equally likely, we can learn a model locally by maximizing the marginal likelihood node by node.

When there are no missing data, the marginal likelihood $p(\mathcal{D}|M_h)$ can be calculated in closed form under the assumptions that all variables are discrete, or all variables follow Gaussian distributions and the dependencies between children and parents are linear. These two cases are described in the next examples. We conclude by noting that the calculation of the marginal likelihood of the data is the essential component for the calculation of the Bayesian estimate of the parameter θ_h, which is given by the expected value of the posterior distribution:

$$p(\theta_h|\mathcal{D}) = \frac{p(\mathcal{D}|\theta_h)p(\theta_h)}{p(\mathcal{D}|M_h)} = \prod_i \frac{p(\mathcal{D}|\theta_{hi})p(\theta_{hi})}{p(\mathcal{D}|M_{hi})}.$$

Example 2.1 (Discrete Variable Networks) Suppose the variables $Y_1, \ldots, Y_v$ are all discrete, and denote by c_i the number of categories of Y_i. The dependency of each variable Y_i on its parents is represented by a set of *multinomial distributions* that describe the conditional distribution of Y_i on the configuration j of the parent variables $Pa(Y_i)$. This representation leads to writing the likelihood function as:

$$p(\mathcal{D}|\theta_h) = \prod_{ijk} \theta_{ijk}^{n_{ijk}}$$

where the parameter θ_{ijk} denotes the conditional probability $p(y_{ik}|pa(y_i)_j)$; n_{ijk} is the sample frequency of $(y_{ik}, pa(y_i)_j)$, and $n_{ij} = \sum_k n_{ijk}$ is the marginal frequency of $pa(y_i)_j$. Figure 4 shows an example of the notation for a network with three variables. With the data in this example, the likelihood function is written as:

$$\{\theta_{11}^4\theta_{12}^3\}\{\theta_{21}^3\theta_{22}^4\}\{\theta_{311}^1\theta_{312}^1 \times \theta_{321}^1\theta_{322}^0 \times \theta_{331}^2\theta_{332}^0 \times \theta_{341}^1\theta_{342}^1\}.$$

The first two terms in the products are the contributions of nodes Y_1 and Y_2 to the likelihood, while the last product is the contribution of the node Y_3, with terms corresponding to the four conditional distributions of Y_3 given each of the four parent configurations.

The *hyper Dirichlet distribution* with parameters α_{ijk} is the conjugate Hyper Markov law [21] and it is defined by a density function proportional to the product $\prod_{ijk} \theta_{ijk}^{\alpha_{ijk}-1}$. This distribution encodes the assumption that the parameters θ_{ij} and $\theta_{i'j'}$ are independent for $i' \neq i$ and $j \neq j'$. These assumptions are

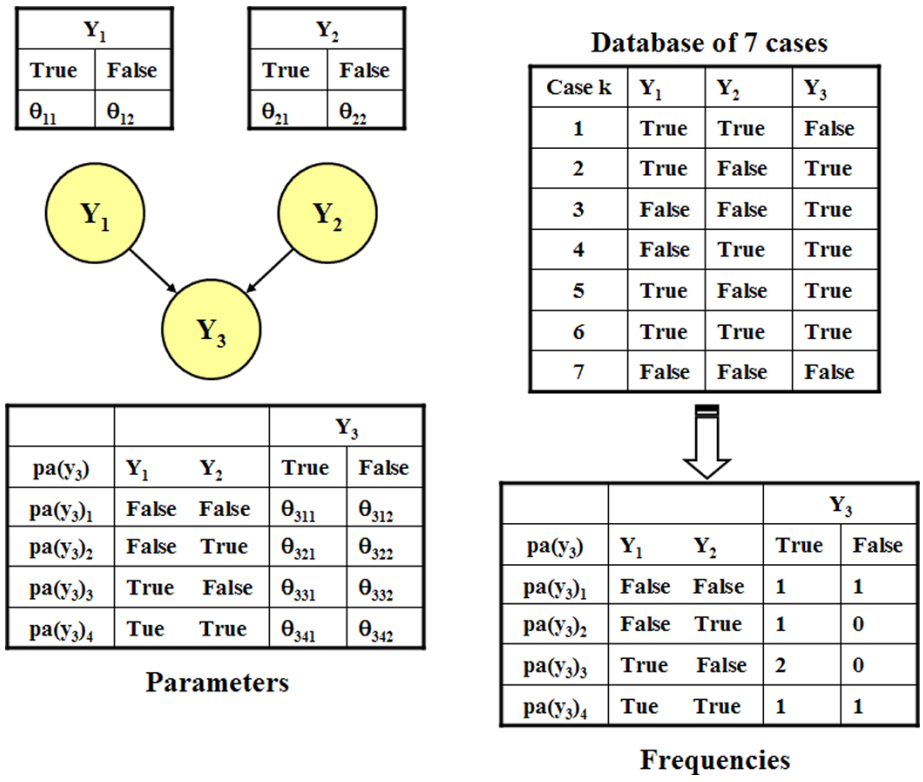


Figure 4: *A simple Bayesian network describing the dependency of Y_3 on Y_1 and Y_2 that are marginally independent. The table on the left describes the parameters θ_{3jk} ($j = 1, \ldots, 4$ and $k = 1, 2$) used to define the conditional distributions of $Y_3 = y_{3k}|pa(y_3)_j$, assuming all variables are binary. The two tables on the right describe a simple database of seven cases, and the frequencies n_{3jk}. The full joint distribution is defined by the parameters θ_{3jk}, and the parameters θ_{1k} and θ_{2k} that specify the marginal distributions of Y_1 and Y_2.*

known as *global and local parameter independence* [97], and are valid only under the assumption the hyper-parameters α_{ijk} satisfy the consistency rule $\sum_j \alpha_{ij} = \alpha$ for all i [40, 34]. Symmetric Dirichlet distributions satisfy easily this constraint by setting $\alpha_{ijk} = \alpha/(c_i q_i)$ where q_i is the number of states of the parents of Y_i. One advantage of adopting symmetric hyper Dirichlet priors in model selection is that, if we fix α constant for all models, then the comparison of posterior probabilities of different models is done conditionally on the same quantity α. With these parameterization and choice of prior distributions, the marginal likelihood is given by the equation

$$\prod_i p(\mathcal{D}|M_{hi}) = \prod_{ij} \frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij} + n_{ij})} \prod_k \frac{\Gamma(\alpha_{ijk} + n_{ijk})}{\Gamma(\alpha_{ijk})}$$

where $\Gamma(\cdot)$ denotes the Gamma function, and the Bayesian estimate of the parameter θ_{ijk} is the posterior mean

$$E(\theta_{ijk}|\mathcal{D}) = \frac{\alpha_{ijk} + n_{ijk}}{\alpha_{ij} + n_{ij}}. \tag{3}$$

More details are in [73].

Example 2.2 (Linear Gaussian Networks) Suppose now that the variables $Y_1, \ldots, Y_v$ are all continuous, and the conditional distribution of each variable Y_i given its parents $Pa(y_i) \equiv \{Y_{i1}, \ldots, Y_{ip(i)}\}$ follows a *Gaussian distribution* with mean that is a linear function of the parent variables, and conditional variance $\sigma_i^2 = 1/\tau_i$. The parameter τ_i is called the precision. The dependency of each variable on its parents is represented by the linear regression equation:

$$\mu_i = \beta_{i0} + \sum_j \beta_{ij} y_{ij}$$

that models the conditional mean of Y_i given the parent values y_{ij}. Note that the regression equation is additive (there are no interactions between the parent variables) to ensure that the model is graphical [57]. In this way, the dependency of Y_i on a parent Y_{ij} is equivalent to having the regression coefficient $\beta_{ij} \neq 0$. Given a set of exchangeable observations $\mathcal{D}$, the likelihood function is:

$$p(\mathcal{D}|\theta_h) = \prod_i (\tau_i/(2\pi))^{n/2} \prod_k \exp\left[-\tau_i (y_{ik} - \mu_{ik})^2/2\right]$$

where μ_{ik} denotes the value of the conditional mean of Y_i, in case k, and the vector θ_h denotes the set of parameters τ_i, β_{ij}. It is usually more convenient to use a matrix notation and we use the $n \times (p(i) + 1)$ matrix X_i to denote the matrix of regression coefficients, with kth row given by $(1, y_{i1k}, y_{i2k}, \ldots, y_{ip(i)k})$, β_i to denote the vector of parameters $(\beta_{i0}, \beta_{i1}, \ldots, \beta_{ip(i)})^T$ associated with Y_i and, in this example, y_i to denote the vector of observations $(y_{i1}, \ldots, y_{in})^T$. With this notation, the likelihood can be written in a more compact form:

$$p(\mathcal{D}|\theta_h) = \prod_i (\tau_i/(2\pi))^{n/2} \exp\left[-\tau_i (y_i - X_i\beta_i)^T (y_i - X_i\beta_i)/2\right]$$

There are several choices to model the prior distribution on the parameters τ_i and β_i. For example, the conditional variance can be further parameterized as:

$$\sigma_i^2 = V(Y_i) - cov(Y_i, Pa(y_i)) V(Pa(y_i))^{-1} cov(Pa(y_i), Y_i)$$

where $V(Y_i)$ is the marginal variance of Y_i, $V(Pa(y_i))$ is the variance-covariance matrix of the parent variables, and $cov(Y_i, Pa(y_i))$ ($cov(Pa(y_i), Y_i)$) is the row (column) vector of covariances between Y_i and each

parent Y_{ij}. With this parameterization, the prior on τ_i is usually a hyper-Wishart distribution for the joint variance-covariance matrix of $Y_i, Pa(y_i)$ [17]. The Wishart distribution is the multivariate generalization of a Gamma distribution. An alternative approach is to work directly with the conditional variance of Y_i. In this case, we estimate the conditional variances of each set of parents-child dependency and then the joint multivariate distribution that is needed for the reasoning algorithms is derived by multiplication. More details are described for example in [105] and [33].

We focus on this second approach and again use the global parameter independence [97] to assign independent prior distributions to each set of parameters τ_i, β_i that quantify the dependency of the variable Y_i on its parents. In each set, we use the standard hierarchical prior distribution that consists of a marginal distribution for the precision parameter τ_i and a conditional distribution for the parameter vector β_i, given τ_i. The standard conjugate prior for τ_i is a *Gamma distribution*

$$\tau_i \sim \text{Gamma}(\alpha_{i1}, \alpha_{i2}) \;\; p(\tau_i) = \frac{1}{\alpha_{i2}^{\alpha_{i1}}\Gamma(\alpha_{i1})}\tau_i^{\alpha_{i1}-1}e^{-\tau_i/\alpha_{i2}}$$

where

$$\alpha_{i1} = \frac{\nu_{io}}{2}, \quad \alpha_{i2} = \frac{2}{\nu_{io}\sigma_{io}^2}.$$

This is the traditional Gamma prior for τ_i with hyper-parameters ν_{io} and σ_{io}^2 that can be given the following interpretation. The marginal expectation of τ_i is $E(\tau_i) = \alpha_{i1}\alpha_{i2} = 1/\sigma_{io}^2$ and

$$E(1/\tau_i) = \frac{1}{(\alpha_{i1}-1)\alpha_{i2}} = \frac{\nu_{io}\sigma_{io}^2}{\nu_{io}-2}$$

is the prior expectation of the population variance. Because the ratio $\nu_{io}\sigma_{io}^2/(\nu_{io}-2)$ is similar to the estimate of the variance in a sample of size ν_{io}, σ_{io}^2 is the prior population variance, based on ν_{io} cases seen in the past. Conditionally on τ_i, the prior density of the parameter vector β_i is supposed to be multivariate Gaussian:

$$\beta_i|\tau_i \sim N(\beta_{io}, (\tau_i R_{io})^{-1})$$

where $\beta_{io} = E(\beta_i|\tau_i)$. The matrix $(\tau_i R_{io})^{-1}$ is the prior variance-covariance matrix of $\beta_i|\tau_i$ and R_{io} is the identity matrix so that the regression coefficients are a priori independent, conditionally on τ_i. The density function of β_i is

$$p(\beta_i|\tau_i) = \frac{\tau_i^{(p(i)+1)/2}\det(R_{io})^{1/2}}{(2\pi)^{(p(i)+1)/2}}e^{-\tau_i/2(\beta_i-\beta_{io})^T R_{io}(\beta_i-\beta_{io})}.$$

With this prior specifications, it can be shown that the marginal likelihood $p(\mathcal{D}|M_h)$ can be written in product form $\prod_i p(\mathcal{D}|M_{hi})$, where each factor is given by the quantity:

$$p(\mathcal{D}|M_{hi}) = \frac{1}{(2\pi)^{n/2}}\frac{\det R_{io}^{1/2}}{\det R_{in}^{1/2}}\frac{\Gamma(\nu_{in}/2)}{\Gamma(\nu_{io}/2)}\frac{(\nu_{io}\sigma_{io}^2/2)^{\nu_{io}/2}}{(\nu_{in}\sigma_{in}^2/2)^{\nu_{in}/2}}$$

and the parameters are specified by the next updating rules:

$$\begin{array}{lcl}
\alpha_{i1n} & = & \nu_{io}/2 + n/2 \\
1/\alpha_{i2n} & = & (-\beta_{in}^T R_{in}\beta_{in} + y_i^T y_i + \beta_{io}^T R_{io}\beta_{io})/2 + 1/\alpha_{i2} \\
\nu_{in} & = & \nu_{io} + n \\
\sigma_{in} & = & 2/(\nu_{in}\alpha_{i2n}) \\
R_{in} & = & R_{io} + X_i^T X_i \\
\beta_{in} & = & R_{in}^{-1}(R_{io}\beta_{io} + X_i^T y_i)
\end{array}$$

The Bayesian estimates of the parameters are given by the posterior expectations:

$$E(\tau_i|y_i) = \alpha_{i1n}\alpha_{i2n} = 1/\sigma^2_{in}, \quad E(\beta_i|y_i) = \beta_{in},$$

and the estimate of σ^2_i is $\nu_{in}\sigma^2_{in}/(\nu_{in}-2)$. More controversial is the use of improper prior distributions that describe lack of prior knowledge about the network parameters by uniform distributions [66]. In this case, we set $p(\beta_i, \tau_i) \propto \tau_i^{-c}$, so that $\nu_{io} = 2(1-c)$ and $\beta_{io} = 0$. The updated hyper-parameters are:

$$\begin{array}{lcl} \nu_{in} & = & \nu_{io} + n \\ R_{in} & = & X_i^T X_i \\ \beta_{in} & = & (X_i^T X_i)^{-1} X_i^T y_i \quad \text{least squares estimate of } \beta \\ \sigma_{in} & = & RSS_i/\nu_{in} \\ RSS_i & = & y_i^T y_i - y_i^T X_i (X_i^T X_i)^{-1} X_i^T y_i \quad \text{residual sum of squares} \end{array}$$

and the marginal likelihood of each local dependency is

$$p(\mathcal{D}|M_{hi}) = \frac{1}{(2\pi)^{(n-p(i)-1)/2}} \Gamma((n-p(i)-2c+1)/2)(RSS_i/2)^{-(n-p(i)-2c+1)/2} \frac{1}{\det(X_i^T X_i)^{1/2}}.$$

A very special case is $c = 1$ that corresponds to $\nu_{io} = 0$. In this case, the local marginal likelihood simplifies to

$$p(\mathcal{D}|M_{hi}) = \frac{1}{(2\pi)^{(n-p(i)-1)/2}} \Gamma((n-p(i)-1)/2)(RSS_i/2)^{-(n-p(i)-1)/2} \frac{1}{\det(X_i^T X_i)^{1/2}}.$$

The estimates of the parameters σ_i and β_i become the traditional least squares estimates $RSS_i/(\nu_{in}-2)$ and β_{in}. This approach can be extended to model an unknown variance-covariance structure of the regression parameters, using Normal-Wishart priors [33]

2.2.2 Model Search

The likelihood modularity allows local model selection and simplifies the complexity of model search. Still, the space of the possible sets of parents for each variable grows exponentially with the number of candidate parents and successful heuristic search procedures (both deterministic and stochastic) have been proposed to render the task feasible [16, 55, 96, 108]. The aim of these heuristic search procedures is to impose some restrictions on the search space to capitalize on the decomposability of the posterior probability of each Bayesian network M_h. One suggestion, put forward by [16], is to restrict the model search to a subset of all possible networks that are consistent with an ordering relation $\succ$ on the variables $\{Y_1, ..., Y_v\}$. This ordering relation $\succ$ is defined by $Y_j \succ Y_i$ if Y_i cannot be parent of Y_j. In other words, rather than exploring networks with arcs having all possible directions, this order limits the search to a subset of networks in which there is only a subset of directed associations. At first glance, the requirement for an order among the variables could appear to be a serious restriction on the applicability of this search strategy, and indeed this approach has been criticized in the artificial intelligence community because it limits the automation of model search. From a modeling point of view, specifying this order is equivalent to specifying the hypotheses that need to be tested, and some careful screening of the variables in the data set may avoid the effort to explore a set of not sensible models. For example, we have successfully applied this approach to model survey data [87, 89] and more recently genotype data [90]. Recent results have shown that restricting the search space by imposing an order among the variables yields a more regular space over the network structures [30].

In functional genomics, the determination of this order can be aided by the available information about gene control interactions embedded into known pathways. When the variables represent gene products, such as gene expression data, the order relationship can describe known regulatory mechanisms and it has been exploited for example in [92] to restrict the set of possible dependency structures between genes. This ordering operation can be largely automated by using some available programs, such as MAPPFinder [24]

or GenMAPP [18], able to automatically map gene expression data to known pathways. For genes with unknown function, one can use different orders with random restarts. Other search strategies based on genetic algorithms [55], "ad hoc" stochastic methods [96] or Markov Chain Monte Carlo methods [30] can also be used. An alternative approach to limit the search space is to define classes of equivalent directed graphical models [13].

The order imposed on the variables defines a set of candidate parents for each variable Y_i and one way to proceed is to implement an independent model selection for each variable Y_i and then link together the local models selected for each variable Y_i. A further reduction is obtained using the greedy search strategy deployed by the *K2 algorithm* [16]. The K2 algorithm is a bottom-up strategy that starts by evaluating the marginal likelihood of the model in which Y_i has no parents. The next step is to evaluate the marginal likelihood of each model with one parent only and if the maximum marginal likelihood of these models is larger than the marginal likelihood of the independence model, the parent that increases the likelihood most is accepted and the algorithm proceeds to evaluate models with two parents. If none of the models has marginal likelihood that exceeds that of the independence model, the search stops. The K2 algorithm is implemented in Bayesware Discoverer (http://www.bayesware.com), and the R-package Deal [4]. Greedy search can be trapped in local maxima and induce spurious dependency and a variant of this search to limit spurious dependency is stepwise regression [62]. However, there is evidence that the K2 algorithm performs as well as other search algorithms [107].

2.2.3 Validation

The automation of model selection is not without problems and both diagnostic and predictive tools are necessary to validate a multivariate dependency model extracted from data. There are two main approaches to model validation: one addresses the *goodness of fit* of the network selected from data and the other assesses the *predictive accuracy* of the network in some predictive/diagnostic tests.

The intuition underlying goodness of fit measures is to check the accuracy of the fitted model versus the data. In regression models in which there is only one dependent variable, the goodness of fit is typically based on some summary of the residuals that are defined by the difference between the observed data and the data reproduced by the fitted model. Because a Bayesian network describes a multivariate dependency model in which all nodes represent random variables, we developed *blanket residuals* [86] as follows. Given the network induced from data, for each case k in the database we compute the values fitted for each node Y_i, given all the other values. Denote this fitted value by $\hat{y}_{ik}$ and note that, by the global Markov property, only the configuration in the Markov blanket of the node Y_i is used to compute the fitted value. For categorical variables, the fitted value $\hat{y}_{ik}$ is the most likely category of Y_i given the configuration of its Markov blanket, while for numerical variables the fitted value $\hat{y}_{ik}$ can be either the expected value of Y_i, given the Markov blanket, or the modal value. In both cases, the fitted values are computed by using one of the algorithms for probabilistic reasoning described in Section 2. By repeating this procedure for each case in the database, we compute fitted values for each variable Y_i, and then define the blanket residuals by

$$r_{ik} = y_{ik} - \hat{y}_{ik}$$

for numerical variables, and by

$$c_{ik} = \delta(y_{ik}, \hat{y}_{ik})$$

for categorical variables, where the function $\delta(a, b)$ takes value $\delta = 0$ when $a = b$ and $\delta = 1$ when $a \neq b$. Lack of significant patterns in the residuals r_{ik} and approximate symmetry about 0 will provide evidence in favor of a good fit for the variable Y_i, while anomalies in the blanket residuals can help to identify weaknesses in the dependency structure that may be due to outliers or leverage points. Significance testing of the goodness of fit can be based on the standardized residuals:

$$R_{ik} = \frac{r_{ik}}{\sqrt{V(y_i)}}$$

where the variance $V(y_i)$ is computed from the fitted values. Under the hypothesis that the network fits the data well, we would expect to have approximately 95% of the standardized residuals within the limits [-2,2]. When the variable Y_i is categorical, the residuals c_{ik} identify the error in reproducing the data and can be summarized to compute the error rate for fit.

Because these residuals measure the difference between the observed and fitted values, anomalies in the residuals can identify inadequate dependencies in the networks. However, residuals that are on average not significantly different from 0 do not necessarily prove that the model is good. A better validation of the network should be done on an independent test set to show that the model induced from one particular data set is *reproducible* and gives good predictions. Measures of the predictive accuracy can be the monitors based on the *logarithmic scoring function* [39]. The basic intuition is to measure the degree of surprise in predicting that the variable Y_i will take a value y_{ih} in the hth case of an independent test set. The measure of surprise is defined by the score

$$s_{ih} = -\log p(y_{ih}|MB(y_i)_h)$$

where $MB(y_i)_h$ is the configuration of the Markov blanket of Y_i in the test case h, $p(y_{ih}|MB(y_i)_h)$ is the predictive probability computed with the model induced from data, and y_{ih} is the value of Y_i in the hth case of the test set. The score s_{ih} will be 0 when the model predicts y_{ih} with certainty, and increases as the probability of y_{ih} decreases. The scores can be summarized to derive *local and global monitors* and to define tests for predictive accuracy [17].

In the absence of an independent test set, standard cross validation techniques are typically used to assess the predictive accuracy of one or more nodes [41]. In K-fold cross validation, the data are divided into K non-overlapping sets of approximately the same size. Then $K-1$ sets are used for retraining (or inducing) the network from data that is then tested on the remaining set using monitors or other measures of the predictive accuracy [42]. By repeating this process K times, we derive independent measures of the predictive accuracy of the network induced from data as well as measures of the robustness of the network to sampling variability. Note that the predictive accuracy based on cross-validation is usually an over-optimistic measure, and several authors have recently argued that cross-validation should be used with caution [5], particularly with small sample sizes.

3 Genomic Applications of Bayesian Networks

Bayesian networks have been applied to the analysis of several gene products, including gene expression measured with microarrays [28, 106] and proteins [46]. This section describes some applications of Bayesian networks in genomics. In the first two sections we use Bayesian networks to model the complex structure of gene-gene interactions in complex traits, using genetic markers and gene expression data measured with microarrays. The last section shows an application of Bayesian networks to proteomics. In all applications, the study design was a *case-control* [82] with subjects selected according to their disease status: cases are subjects affected with the particular disease of interest, while controls are unaffected with the disease.

3.1 Networks of Genetic Markers

Many complex diseases are likely to be determined by the joint action of particular genotypes and their interaction with environmental factors. Alzheimer disease is an example of a complex trait related to multiple genes and there is evidence that several genes and the environment influence the risk of this disease [82]. Another example is diabetes mellitus, for which several studies have identified different genotypes that are

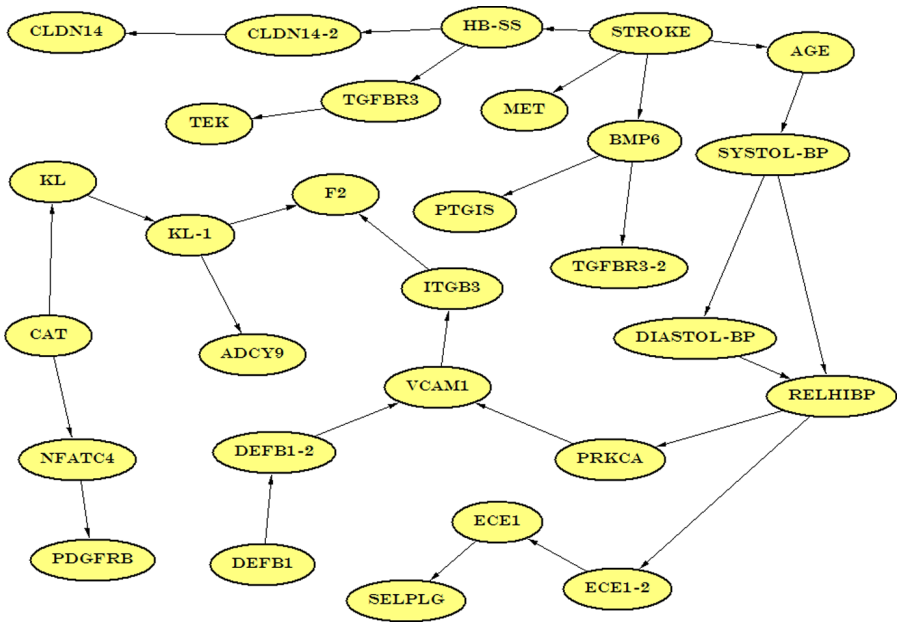


Figure 5: *A Bayesian network representing a complex trait given by the interaction of several genes and clinical covariates.*

associated with the disease [2]. In both examples, polymorphic loci of several genes have been found to be associated with the disease.

It is well known that the majority of the DNA sequence is equal across all individuals except for a small proportion of positions that have more than one form (*allele*). A piece of DNA that has more than one form, each occurring with at least 1% frequency in the population, is called *polymorphic* and when the piece is a single base of the DNA, it is called a single nucleotide polymorphism (SNP). SNPs work as flags on a high density map of the human genome and allow us to identify those genes whose polymorphisms may be causative of the disease [52]. In case-control studies, the data available for this discovery process are typically genotypes of case and control subjects at polymorphic loci, together with information about several clinical covariates and environmental factors. The genotype can be coded as either the presence/absence of the minor allele (the allele with smaller frequency in the population) in the two loci of the chromosome pair, or as the complete allele pair that can be homozygous for the major allele, homozygous for the minor allele, or heterozygous when the two alleles are different.

The discovery of complex gene-environment interactions that confer susceptibility to disease requires advanced multivariate modeling tools and a typical solution is to resort to logistic regression models to describe the odds for the disease given a particular genotype. The advantages of logistic regression models are that they can be used to assess whether the association between the risk for disease and a particular genotype is *confounded* by some external factor (such as population admixture [67]) and they can be used to test whether an external factor or a particular genotype is an *effect modifier* of an association [47]. However, logistic regression models pose three serious limitations: when the susceptibility to disease is caused by the interaction among several genes, the number of parameters required to fit a logistic regression model increases at an exponential rate; the genotypes are treated as covariates rather than random variables; logistic regression is limited to examine the association between one phenotypic character at a time. To simultaneously overcome these three limitations, we have recently proposed to use Bayesian networks to discover the genetic makeup

that confers susceptibility to overt stroke in patients with sickle cell anemia.

The complications of sickle cell anemia are likely to be determined by the actions of genes that modify the pathophysiology initiated by sickle hemoglobin. Overt stroke (CVA) occurs in about 10% of patients with sickle cell anemia. To define the genetic basis of CVA in sickle cell anemia we examined the association of SNPs in several candidate genes of different functional classes with the likelihood of CVA. In our study, we considered 92 patients with a confirmed history of or incident complete non-hemorrhagic CVA, documented by imaging studies and 453 controls (patients who did not have a stroke in five years follow up). We modeled the genetic markers and their association with the CVA phenotype by Bayesian networks using the structural learning approach described in Section 2.2. We validated the network of association induced from data using cross-validation, which showed that the network of gene-gene-phenotype interaction can predict the likelihood of CVA in patients with sickle cell anemia with 99.7% accuracy. We also validated the model using an independent set of 114 individuals with an accuracy of 98%. In both tests, the accuracy was measured by the frequency of individuals for whom the Bayesian network model predicted the correct phenotype with probability above 0.5 [91]. With this approach, we discovered a network of interacting genes that may confer susceptibility to CVA in patients with sickle cell anemia. Part of the network is displayed in Figure 5 and identifies polymorphisms of the genes MET and BMP6 as directly associated with CVA. The Markov blanket of the node representing the phenotype (Stroke) identifies the gene-gene-environment interaction that confers susceptibility to the disease. It consists of polymorphisms of the genes MET and BMP6, age of the patient and whether or not the patient is affected by α-thalassemia (node HB-SS). Dependencies between polymorphisms of other genes may be interpreted as an effect of population admixture, while dependencies between polymorphism of the same gene denote linkage disequilibrium [67].

3.2 Gene Expression Networks

The coherent probabilistic framework of Bayesian networks can be used not only to model genotype data but also gene expression data. Compared to standard expression profiling methods, Bayesian networks are able to represent the directionality of the influence among gene expression and they have been already deployed to understand both gene expression [31] and protein-protein interactions [46].

Another area of application of Bayesian networks in functional genomics is modeling differential expression in comparative experiments. Typical statistical techniques used to identify genes that have differential expression in two or more conditions work assuming that genes act independently [83]. Bayesian networks can be used to identify genes with differential expression by simultaneously modeling the structure of gene-gene interaction. An example is in Figure 6 that describes a network of gene expression interaction learned from a case control study of prostate cancer. We used a data set of expression profiles derived from 102 prostatectomy specimens. Cases were 52 cancer specimens of patients undergoing surgery between 1996 and 1997, and controls were 50 normal specimens. The expression profiles were derived with the U95Av2 Affymetrix microarray and are described in [95]. We first analyzed the data with BADGE [88], a program for differential analysis that uses Bayesian model averaging to compute the posterior probability of differential expression. We selected about 200 genes with very large probability of differential expression and then modeled the network of interaction of gene expression. We used information about known functions of some genes to limit the search space and, for example, imposed the restriction that genes known as transcription factors could only be tested as parents of all other nodes. In the absence of an independent set, the final network was tested with 5-fold cross validation and had 93% accuracy in predicting the clinical status, and an average accuracy of about 80% in predicting the expression of each gene given the others.

Besides the identification of a molecular profile based on those genes that are directly related to the clinical status, the network displays some interesting associations. For example, changes in expression of TRCγ (41468_AT: a known enhancer of transcriptional activity specific for prostatic adenocarcinoma cell line) are associated with changes of expression of several genes including SIM2(39608_AT: a transcription repression); PSMA (1740_G_AT: a gene associated with prostate cancer) and MRP (36174_AT: a gene known as potential

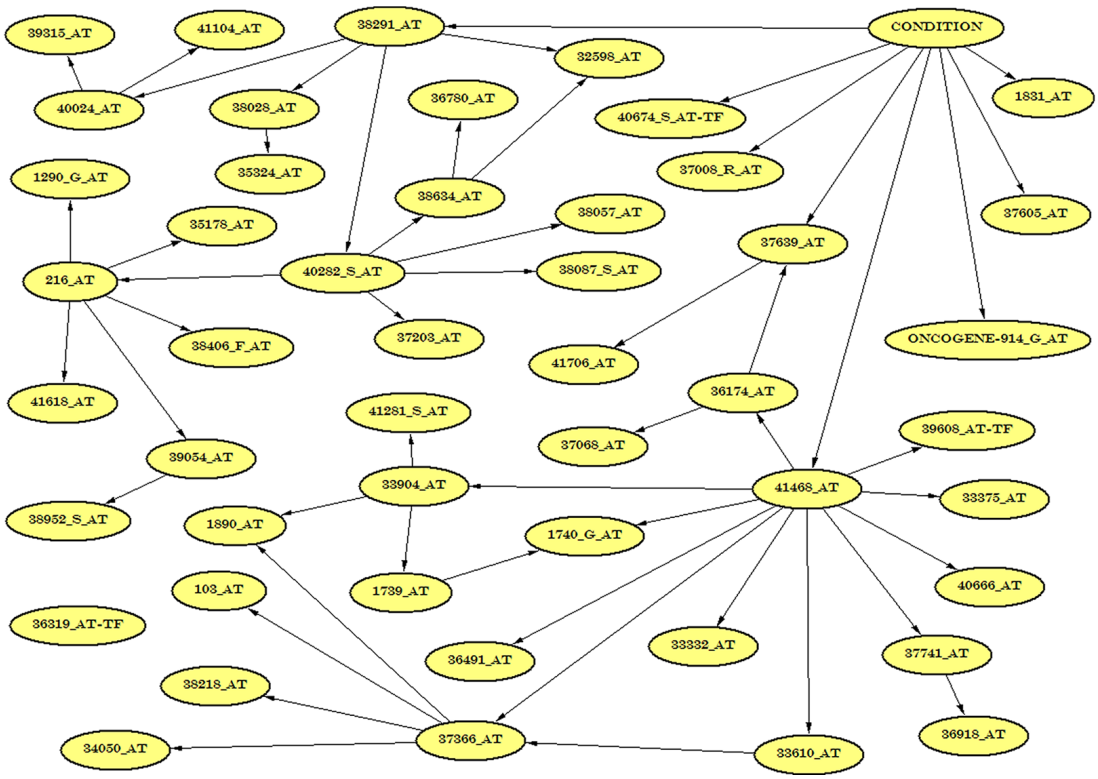


Figure 6: *A Bayesian network associating genes that are differentially expressed between normal and tumor specimens (node Condition). Genes are labelled by the Affymetrix probe ID.*

predictor of chemotherapy response). The probability of changes in expression of Hepsin (37639_AT: a gene with cell growth function) depends on both the clinical status and changes in expression of MRP and differential expression of the Hepsin gene influences changes in expression of AMACR (41706_AT: a marker of tumor differentiation known to be essential for growth of prostate cancer [23, 26].) These directed associations suggest a mechanism by which changes in the transcription factor TRCγ influence changes in genes involved in tumor growth. Another interesting fact is the directed association between Adipsin (40282_S_AT: a gene supposed to have a role in immune system biology) and CRBP1 (38634_AT: a gene known to contribute to cancer by disrupting the vitamin A metabolism). One theory is that cancer arises from the accumulation of genetic changes that induce unlimited, self-sufficient growth and resistance to normal regulatory mechanisms and these two sets of dependencies are consistent with this conjecture.

Of course, the nature of the data collected in a case-control study limits the dependency structure to represent associations rather than causal effects. This limitation is due to the data rather than the modeling approach and data produced by controlled experiments have been used to induce causal networks [99, 106]. We will discuss this issue further in Section 4.3.

3.3 In Silico Integrative Genomics

The predictive capabilities of Bayesian networks can be deployed for in silico identification of unobserved characteristics of the genome. Genetic studies are designed to identify regions of the genome associated with a disease phenotype. The success rate of these studies could be improved if we were able to predict in advance, before conducting the study, the likelihood of a SNP or a mutation in a particular region to be indeed pathogenic. To do so, we need to integrate the information available about SNPs and mutations with the available information about proteins, and predict that a particular change in the DNA will actually lead to a change in the encoded protein. Using Bayesian networks, we have developed a novel algorithm to predict pathogenic single amino acid changes, either non-synonymous SNPs (nsSNPs) — SNPs causing a change in the encoded amino acid — or missense mutations, in conserved protein domains [8]. We found that the probability of a microbial missense mutation causing a change in phenotype depended on how much difference it made in several phylogenetic, biochemical, and structural features related to the single amino acid substitution. We tested our model on pathogenic allelic variants (missense mutations or nsSNPs) included in OMIM (www.ncbi.nlm.nih.gov/omim) and on the other nsSNPs in the same genes from dbSNP (www.ncbi.nlm.nih.gov/SNP) as the non-pathogenic variants. Our results show that our model was able to predict pathogenic variants with a 10% false-positive rate.

4 Advanced Topics

This section describes some extensions of Bayesian networks to classification and for modeling nonlinear and temporal dependencies.

4.1 Bayesian Networks and Classification

The goal of many studies in genomics medicine is the discovery of a molecular profile for disease diagnosis or prognosis. The molecular profile is typically based on gene expression [38, 68, 104]. Bayesian networks have been used in the past few years as supervised classification models able to discover and represent molecular profiles that characterize a disease [49, 109]. This section describes particular classification models that are simple Bayesian networks.

4.1.1 Classification

The term "supervised classification" covers two complementary tasks: the first is to identify a function mapping a set of *attributes* onto a *class*, and the other is to assign a class label to a set of unclassified cases described by attribute values. We denote by C the variable whose states represent the class labels c_i, and by Y_i the attributes. In our context, the class variable may represent a clinical status, and the attributes can be gene products such as gene expression data or genotypes.

Classification is typically performed by first training a classifier on a set of labelled cases (*training set*) and then using it to label unclassified cases (*test set*). The supervisory component of this classifier resides in the training signal, which provides the classifier with a way to assess a dependency measure between attributes and classes. The classification of a case with attribute values $y_{1k}, \ldots, y_{vk}$ is then performed by computing the probability distribution $p(C \mid y_{1k}, \ldots, y_{vk})$ of the class variable, given the attribute values, and by labelling the case with the most probable label. Most of the algorithms for learning classifiers described as Bayesian networks impose a restriction on the network structure, namely that there cannot be arcs pointing to the class variable. In this case, by the local Markov property, the joint probability $p(y_{1k}, \ldots, y_{vk}, c_k)$ of class and attributes is factorized as $p(c_k)p(y_{1k}, \ldots, y_{vk} \mid c_k)$. The simplest example is known as a *Naïve Bayes* (NB) classifier [25, 53], and makes the further simplification that the attributes Y_i are conditionally independent given the class C so that

$$p(y_{1k}, \ldots, y_{vk}|c_k) = \prod_i p(y_{ik}|c_k).$$

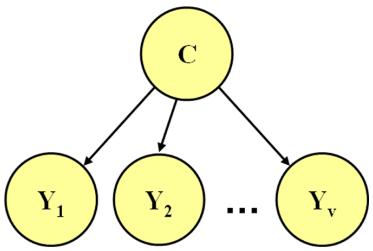


Figure 7: *The structure of the Naïve Bayes classifier*

Figure 7 depicts the directed acyclic graph of a NB classifier. Because of the restriction on the network topology, the training step for a NB classifier consists of estimating the conditional probability distributions of each attribute, given the class, from a training data set. When the attributes are discrete or continuous variables and follow Gaussian distributions, the parameters are learned by using the procedure described in Section 2.2. Once trained, the NB classifies a case by computing the posterior probability distribution over the classes via Bayes' Theorem and assigns the case to the class with the highest posterior probability.

Other classifiers have been proposed to relax the assumption that attributes are conditionally independent given the class. Perhaps the most competitive one is the *Tree Augmented Naïve Bayes*(TAN) classifier [29] in which all the attributes have the class variable as a parent as well as another attribute. To avoid cycles, the attributes have to be ordered and the first attribute does not have other parents beside the class variable. Figure 8 shows an example of a TAN classifier with five attributes. An algorithm to infer a TAN classifier needs to choose both the dependency structure between attributes and the parameters that quantify this dependency. Due to the simplicity of its structure, the identification of a TAN classifiers does not require any search but rather the construction of a tree among the attributes. An "ad hoc" algorithm called *Construct-TAN* (CTAN) was proposed in [29]. One limitation of the CTAN algorithm to build TAN classifiers is that it applies only to discrete attributes, and continuous attributes need to be discretized.

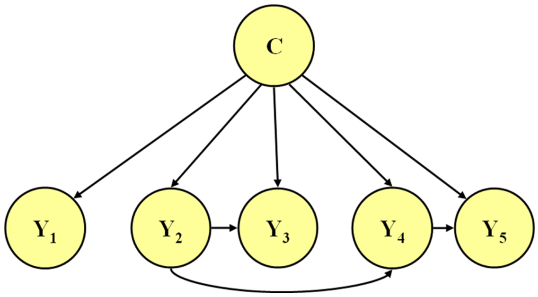


Figure 8: *The structure of a* TAN *classifier*

Other extensions of the NB try to relax some of the assumptions made by the NB or the TAN classifiers. Some examples are the *l-Limited Dependence Bayesian classifier* (l-LDB) in which the maximum number of parents that an attribute can have is l [79]. Another example is the *unrestricted Augmented Naïve Bayes classifier* (ANB) in which the number of parents is unlimited but the scoring metric used for learning, the minimum description length criterion, biases the search toward models with small number of parents per attribute [29]. Due to the high dimensionality of the space of different ANB networks, algorithms that build this type of classifiers must rely on heuristic searches. More examples are reported in [29].

4.1.2 Molecular Classification

Many learning algorithms show a high sensitivity to correlated features. In the case of data sets of gene expression profiles measured with microarrays, the large number of genes must be drastically reduced in order to improve the diagnostic accuracy. Many learning algorithms that build classifiers and perform feature selection have been used in this context [3, 59, 50]. As an example, we used the NB and TAN classifiers to build a molecular classification model using the data set of gene expression measured in prostatectomy specimens (see Figure 6). Table 1, column 2, shows the test accuracy of the classifiers learned by different algorithms that was measured with 5-fold cross validation. The first classifier is a NB and the second classifier is a TAN. In both cases the parameters were learned with the Bayesian approach discussed in Section 2.2. Due the large number of input attributes, we used a filtered version of the wrapped feature selection algorithm described in [54] to increase the predictive accuracy.

Column 3 shows the accuracy of the same classifiers that were built by selecting a subset of the genes and shows that accuracy sensibly increases when feature selection is performed. The genes selected by the feature selection algorithm are 32598_at, 38291_at, 39315_at, 37624_at, 38059_g_at, 36725_at, 31666_f_at, 39417_at, 37639_at and represent a molecular profile for classifying prostatectomy specimens into normal or tumor. Figure 9 shows the TAN structure chosen by the CTAN algorithm with feature selection. It is interesting to note that the selection of genes by the wrapped feature selection differs from those induced by the standard Bayesian algorithm described in Section 2.2. Particularly, neither of the classifiers reaches the classification accuracy of the Bayesian network model in Figure 6.

4.2 Generalized Gamma Networks

Most of the work on learning Bayesian networks from data has focused on learning networks of categorical variables, or networks of continuous variables that are modelled by Gaussian distributions with linear dependencies. However, linearity of the parent-child dependencies and normality of the data are limitations. This section describes a new class of Bayesian networks that addresses these issues.

ALGORITHM	ALL ATTRIBUTES	FEATURE SELECTION
NB	75.4902	87.2549
sCTAN	73.5294	80.3922

Table 1: *Test accuracies for some classifiers without and with feature selection.*

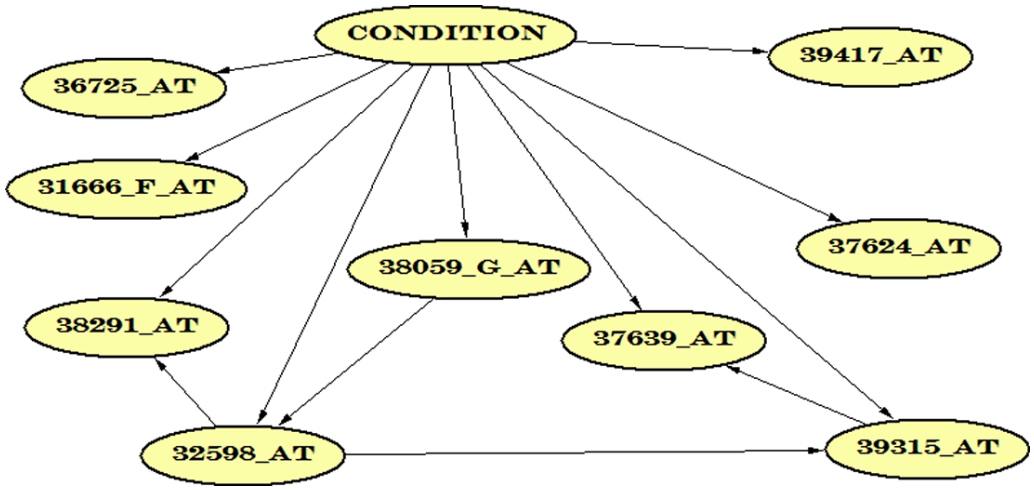


Figure 9: *The structure of the TAN classifier with feature selection in the gene expression dataset.*

4.2.1 Learning and Representation

A feature of gene expression data measured with microarray is the apparent lack of symmetry and there is evidence that they do not follow Gaussian distributions, even after a logarithmic transformation [84]. Figure 10 shows an example. The left histogram shows the density of a sample of 50 expression levels of the Homo sapiens ubiquitin gene in the U95Av2 Affymetrix microarray. The distribution has an exponential decay, with a long right tail. The histogram in the right panel displays the distribution of the log-transformed data and shows the phenomenon that log-transforming the original data removes the right tail but introduces a long left tail. This phenomenon is typically observed when log-transforming data that follow a gamma distribution, with consequent bias induced to estimate the mean [65, Ch 8]. We recently introduced a new class of Bayesian networks called Generalized Gamma networks (GGN) able to describe possibly nonlinear dependencies between variables with non-normal distributions [86]. Compared to other Bayesian network formalisms that have been proposed for representing gene-gene interactions [28], GGNs do not require to discretize gene expression data, or to enforce normality or log-normality assumptions.

In a GGN the conditional distribution of each variable Y_i given the parents $Pa(y_i) = \{Y_{i1}, \ldots, Y_{ip(i)}\}$ follows a Gamma distribution $Y_i|pa(y_i), \theta_i \sim Gamma(\alpha_i, \mu_i(pa(y_i), \beta_i))$, where $\mu_i(pa(y_i), \beta_i)$ is the conditional mean of Y_i and $\mu_i(pa(y_i), \beta_i)^2/\alpha_i$ is the conditional variance. We use the standard parameterization of generalized linear models [65], in which the mean $\mu_i(pa(y_i), \beta_i)$ is not restricted to be a linear function of the parameters β_{ij}, but the linearity in the parameters is enforced in the *linear predictor* η_i, which is itself related to the mean function by the *link function* $\mu_i = g(\eta_i)$. Therefore, we model the conditional density function as:

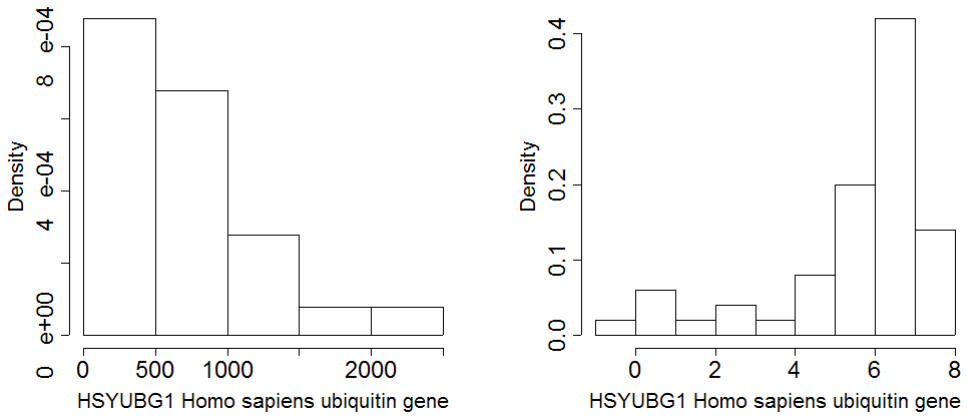


Figure 10: *Distribution of expression data of the HSYUBG1 Homo sapiens ubiquitin gene in a data set of 50 prostatectomy samples measured with the U95Av2 Affymetrix microarray. The left panel shows the histogram of the original expression data. The right panel shows the histogram of the log-transformed gene expression data.*

LINK	$g(\cdot)$	LINEAR PREDICTOR η
IDENTITY	$\mu = \eta$	$\eta_i = \beta_{i0} + \sum_j \beta_{ij} y_{ij}$
INVERSE	$\mu = \eta^{-1}$	$\eta_i = \beta_{i0} + \sum_j \beta_{ij} y_{ij}^{-1}$
LOG	$\mu = e^{\eta}$	$\eta_i = \beta_{i0} + \sum_j \beta_{ij} log(y_{ij})$

Table 2: *Link functions and parameterizations of the linear predictor.*

$$p(y_i|pa(y_i), \theta_i) = \frac{\alpha_i^{\alpha_i}}{\Gamma(\alpha_i)\mu_i^{\alpha_i}} y_i^{\alpha_i - 1} e^{-\alpha_i y_i/\mu_i}, \quad y_i \geq 0 \tag{4}$$

where $\mu_i = g(\eta_i)$ and the linear predictor η_i is parameterized as

$$\eta_i = \beta_{i0} + \sum_j \beta_{ij} f_j(pa(y_i))$$

and $f_j(pa(y_i))$ are possibly nonlinear functions. The linear predictor η_i is a function linear in the parameters β, but it is not restricted to be a linear function of the parent values, so that the generality of Gamma networks is in the ability to encode general non-linear stochastic dependency between the node variables. Table 2 shows example of non-linear mean functions. Figure 11 shows some examples of Gamma density functions, for different shape parameters $\alpha = 1, 1.5, 5$ and mean $\mu = 400$. Note that approximately symmetrical distributions are obtained for particular values of the shape parameter α.

Unfortunately, there is no closed form solution to learn the parameters of a GGN and we have therefore to resort to Markov Chain Monte Carlo methods to compute stochastic estimates [63], or to maximum likelihood to compute numerical approximation of the posterior modes [48]. A well know property of generalized linear

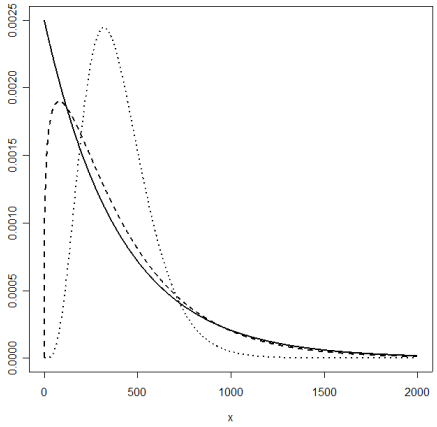


Figure 11: *Example of Gamma density functions for shape parameters $\alpha = 1$ (continuous line), $\alpha = 1.5$ (dashed line), and $\alpha = 5$ (dotted line) and mean $\mu = 400$. For fixed mean, the parameter α determines the shape of the distribution that is skewed to the left for small α and approaches symmetry as α increases.*

models is that the parameters β_{ij} can be estimated independently of α_i, which is then estimated conditionally on β_{ij} [65].

To compute the maximum likelihood estimates of the parameters β_{ij} within each family $(Y_i, Pa(y_i))$, we need to solve the system of equations $\partial \log p(\mathcal{D}|\theta_i)/\partial \beta_{ij} = 0$. The Fisher Scoring method is the most efficient algorithm to find the solution of the system of equations. This iterative procedure is a generalization of the Newton Raphson procedure in which the Hessian matrix is replaced by its expected value. This modification speeds up the convergence rate of the iterative procedure that is known for being usually very efficient — it usually converges in 5 steps for appropriate initial values. Details can be found for example in [65].

Once the ML estimates of β_{ij} are known, say $\hat{\beta}_i$, we compute the fitted means $\hat{\mu}_{ik} = g(\hat{\beta}_{i0} + \sum_j \hat{\beta}_{ij} f_j(pa(y_i))$ and use these quantities to estimate the shape parameter α_i. Estimation of the shape parameter in Gamma distributions is an open issue, and authors have suggested several estimators (see for example [65]). Popular choices are the deviance-based estimator that is defined as

$$\tilde{\alpha}_i = \frac{n-q}{\sum_k (y_{ik} - \hat{\mu}_{ik})^2 / \hat{\mu}_{ik}^2}$$

where q is the number of parameters β_{ij} that appear in the linear predictor. The maximum likelihood estimate $\hat{\alpha}_i$ of the shape parameter α_i would need the solution of the equation

$$n + n\log(\alpha_i) + n\frac{\Gamma(\alpha_i)'}{\Gamma(\alpha_i)} + -\sum_k \log(\hat{\mu}_{ik}) + \sum_k \log(y_{ik}) - \sum_i \frac{y_{ik}}{\hat{\mu}_{ik}} = 0$$

with respect to α_i. We have an approximate closed form solution to this equation based on a Taylor expansion that is discussed in [88].

Also the model selection process requires the use of approximation methods. In this case, we use the Bayesian information criterion (BIC) [48] to approximate the marginal likelihood by $2\log p(\mathcal{D}|\hat{\theta}) - n_p \log(n)$ where $\hat{\theta}$ is the maximum likelihood estimate of θ, and n_p is the overall number of parameters in the network. BIC is independent of the prior specification on the model space and trades off goodness of fit — measured

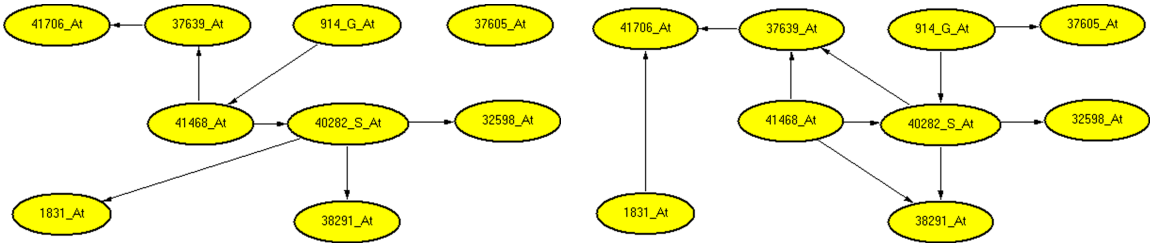


Figure 12: *Gamma networks induced from the 50 normal specimens (left) and 52 tumor specimens (right).*

AFFY-ENTRY	Gene Name	Gene Function
41706_At	alpha-methylacyl-CoA racemase	cellular component
37639_At	Epsin	cell growth
37605_At	COL2A1	collagen
41468_At	TCRγ	cellular defense
914_G_At	ERG	transcription regulation
40282_S_At	Adipsin	role in immune system biology
1831_At	TGFβ	transforming grown factor
38291_At	Human enkephalin gene	signal transduction
32598_A	Nel-like 2	cell growth regulation and differentiation

Table 3: *The nine genes used in the* GGN *and their known functions.*

by the term $2 \log p(\mathcal{D}|\hat{\theta})$ — and model complexity — measured by the term $n_p \log(n)$. We note that BIC factorizes into a product of terms for each variable Y_i and makes it possible to conduct local structural learning.

While the general type of dependencies in Gamma networks makes it possible to model a variety of dependencies within the variables, exact probabilistic reasoning with the network becomes impossible and we need to resort to Gibbs sampling (see Section 2). Our simulation approach uses the adaptative rejection metropolis sampling (ARMS) of [37] when the conditional density $p(y_i|\mathcal{Y}\backslash y_i, \hat{\theta})$ is log-concave, and adaptive rejection with Metropolis sampling in the other cases. See [86] for more details.

4.2.2 An Example

We use a subset of nine of the gene expression data measured from the 102 prostatectomy specimens to show the representation advantages of GGNs. We modeled the dependency structure among the nine genes in the normal and tumor specimens, with an initial order that was chosen by using information about their roles in pathways, when known, and by ranking the remaining genes on the basis of the evidence for differential expression. For example, the gene 914_G_at (ERG) has a transcription regulation function that has been observed in several tumors, so we left this gene high in the order and tested it as parent of all the other nodes. Figure 12 depicts the dependency structures in the two groups. In both cases, we limited the search to dependency models in which the link function was either the identity $\mu = \eta$ or the inverse link $\mu = 1/\eta$. The two network structures were validated by examining the blanket residuals to assess the goodness of fit for each local dependency structure. In both networks we tested whether the standardized blanket residuals had

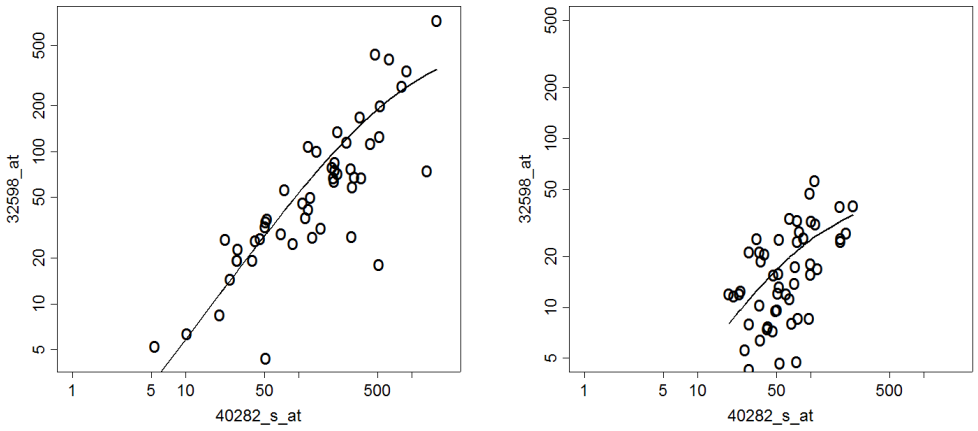
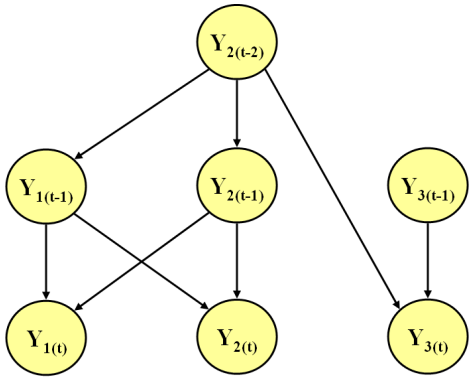


Figure 13: *Scatter plot of the dependency between 40282_S_at and 32598_A in the two* GGNs *induced from the 50 normal specimens (left) and 52 tumor specimens (right). The lines are the dependency fitted by the* GGNs*. Both plots are in log-scales.*

means significantly different from 0 using standard t-tests, and we checked for departures from normality. These tests confirmed the validity of the network structures induced from data, and the correctness of the distributional assumptions.

Evidence of the biological meaning of the dependency structures encoded by the two GGNs gives further support that this technology can help to model complex gene-gene interactions in biological systems. For example, in the network learned from the normal specimens, the gene COL2A1 (37605_at: a collagene) is independent of all other genes, whereas in the network learned from the tumor specimens, this gene is a child of ERG (914_at: an oncogene with transcription regulation functions). Independent studies have associated changes of expression in TGFβ (1831_at: a gene with role in signalling pathways), with changes of expression in COL2A1, and our models suggest a possible mechanism in which this occurs. In the network induced from tumor specimens, TGFβ is directly influencing AMACR (41706_at: a gene known as a marker of tumor differentiation). In both networks, the dependency structure of Adipsin (40282_S_at: a gene supposed to have a role in immune system biology) is essentially the same, besides the fact that Epsin (37639_at: a gene with putative function in cell growth) is independent of Adipsin given TCRγ (41468_at: a gene with role in cell defense) in the network learned from normal specimens. However, even for those genes with the same dependency structure, the probability distributions that quantify these dependencies suggest different gene-gene interactions. Figure 13 shows the smooth, non linear dependency between Adipsin and Nel-like 2 (32598_A) in the two GGNs induced from the 50 normal specimens (left) and 52 tumor specimens (right). The two non linear dependencies show that changes of expression of Adipsin in the network learned from tumor specimens have a much reduced effect on changes of expression of Nel-like 2. As mentioned earlier, one theory is that cancer arises from the accumulation of genetic changes that induce unlimited, self-sufficient growth and resistance to normal regulatory mechanisms. Our different dependency structures suggest that, in the cancer specimens, the gene Adipsin has a weaker control on the gene Nel-like 2 that regulates cell growth and differentiation. The reasonable biological explanation also points out an important feature of GGNs: by modeling gene-gene interaction via non-linear dependency, GGNs can easily describe the biological effect of gene expression saturation, in which gene expression control changes according to changes of expression levels.



		$Y_{3(t)}$	
$Y_{2(t-2)}$	$Y_{3(t-1)}$	-	+
-	-	**0.9**	**0.1**
-	+	**0.7**	**0.3**
+	-	**0.6**	**0.4**
+	+	**0.1**	**0.9**

Figure 14: *A directed acyclic graph that represents the temporal dependency of three categorical variables describing up (+) and down (+) regulations of three genes.*

Alternative dependency structures can suggest new hypothetical pathways, as well as experiments to test putative functions of genes. For example, the propagation of particular expression levels for some genes can identify their impact on the expression level of other genes and provide a platform for *in silico* experiments based on the learned network.

4.3 Bayesian Networks and Temporal Dependency

One of the limitations of Bayesian networks is the inability to represents forward loops: by definition the directed graph that encodes the marginal and conditional independencies between the network variables cannot have cycles. This limitation makes traditional Bayesian networks unsuitable for the representation of many biological systems in which feedback controls are a critical aspect of gene regulation. Dynamic Bayesian networks provide a general framework to integrate multivariate time series of gene products and to represent feed-forward loops and feedback mechanisms [28] that is alternative to other network models of gene regulation [94].

A dynamic Bayesian network is defined by a directed acyclic graph in which nodes continue to represent stochastic variables and arrows represent temporal dependencies that are quantified by probability distributions. The crucial assumption is that the probability distributions of the temporal dependencies are time invariant, so that the directed acyclic graph of a dynamic Bayesian network represents only the necessary and sufficient time transitions to reconstruct the overall temporal process. Figure 14 shows the directed acyclic graph of a dynamic Bayesian network with three variables. The subscript of each node denotes the time lag, so that the arrows from the nodes $Y_{2(t-1)}$ and $Y_{1(t-1)}$ to the node $Y_{1(t)}$ describe the dependency of the probability distribution of the variable Y_1 at time t on the value of Y_1 and Y_2 at time $t-1$. Similarly, the directed acyclic graph shows that the probability distribution of the variable Y_2 at time t is a function of the value of Y_1 and Y_2 at time $t-1$. This symmetrical dependency allows us to represent feedback loops and we used it to describe the regulatory control of glucose in diabetic patients [72]. A dynamic Bayesian network is not restricted to represent temporal dependency of order 1. For example the probability distribution of the variable Y_3 at time t depends on the value of the variable at time $t-1$ as well as the value of the variable Y_2 at time $t-2$. The conditional probability table in Figure 14 shows an example when the variables Y_2, Y_3 are categorical.

By using the local Markov property, the joint probability distribution of the three variables at time t, given the past history $y_{1(t-1)}, \ldots, y_{1(t-l)}, y_{2(t-1)}, \ldots, y_{2(t-l)}, y_{3(t-1)}, \ldots, y_{3(t-l)}$ is given by the product of the three factors:

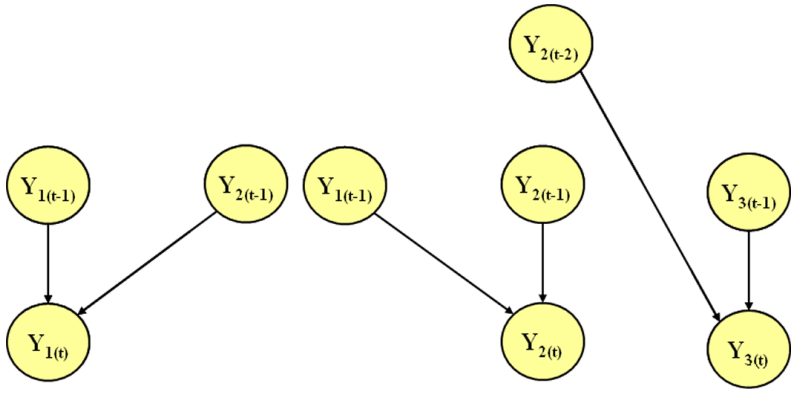


Figure 15: *Modular learning of the dynamic Bayesian network in Figure 14. First a regressive model is learned for each of the three variables at time t, and then the three models are joined by their common ancestors $Y_{1(t-1)}, Y_{2(t-2)}$ and $Y_{2(t-2)}$ to produce the directed acyclic graph in Figure 14.*

$$
\begin{aligned}
p(y_{1(t)}|y_{1(t-1)},\ldots,y_{1(t-l)},y_{2(t-1)},\ldots,y_{2(t-l)},y_{3(t-1)},\ldots,y_{3(t-l)}) &= p(y_{1(t)}|y_{1(t-1)},y_{2(t-1)}) \\
p(y_{2(t)}|y_{1(t-1)},\ldots,y_{1(t-l)},y_{2(t-1)},\ldots,y_{2(t-l)},y_{3(t-1)},\ldots,y_{3(t-l)}) &= p(y_{2(t)}|y_{1(t-1)},y_{2(t-1)}) \\
p(y_{3(t)}|y_{1(t-1)},\ldots,y_{1(t-l)},y_{2(t-1)},\ldots,y_{2(t-l)},y_{3(t-1)},\ldots,y_{3(t-l)}) &= p(y_{3(t)}|y_{3(t-1)},y_{2(t-2)})
\end{aligned}
$$

that represent the probability of transition over time. By assuming that these probability distributions are time invariant, they are sufficient to compute the probability that a process that starts from known values $y_{1(1)}, y_{2(1)}, y_{3(0)}, y_{3(1)}$ evolves into $y_{1(T)}, y_{2(T)}, y_{3(T)}$, by using one of the algorithms for probabilistic reasoning described in Section 2. The same algorithms can be used to compute the probability that a process with values $y_{1(T)}, y_{2(T)}, y_{3(T)}$ at time T started from the initial states $y_{1(1)}, y_{2(1)}, y_{3(0)}, y_{3(1)}$.

Learning dynamic Bayesian networks when all the variables are observable is a straightforward parallel application of the structural learning described in Section 2.2. To build the network, we proceed by selecting the set of parents for each variable Y_i at time t, and then the models are joined by the common ancestors. An example is in Figure 15. The search of each local dependency structure is simplified by the natural ordering imposed on the variables by the temporal frame [32] that constrains the model space of each variable Y_i at time t: the set of candidate parents consists of the variables $Y_{i(t-1)}, \ldots, Y_{i(t-p)}$ as well as the variables $Y_{h(t-j)}$ for all $h \neq i$, and $j = 1, \ldots, p$. The K2 algorithm [16] discussed in Section 2.2 appears to be particularly suitable for exploring the space of dependency for each variable $Y_{i(t)}$. The only critical issue is that the selection of the largest temporal order to explore depends on the sample size, because each temporal lag of order p leads to the loss of the first p temporal observations in the data set [107].

Dynamic Bayesian networks are an alternative approach to represent gene regulatory mechanisms by approximating rates of change described by a system of differential equations with autoregressive models. When the gene products are measured at regularly spaced time points, there is a simple way to approximate the rate of change $dy_{i(t)}/dt = f(y_{gt})$ by a first order linear approximation. This approach has been used to model the rate of change by linear Gaussian networks [22]. However, the development of similar approximations for non regularly spaced time points and for general, non linear, kinetic equations with feedback loops [11] is an open issue. The further advantage of dynamic Bayesian network is to offer an environment for causal inference with well designed temporal experiments.

5 Research Directions

This chapter has discussed the potential usefulness of Bayesian networks to analyze genomic data. However, there are some limitations of the current representation and learning approaches that need further investigation. A main assumption underlying all learning algorithms is that the data are complete, so there are no missing entries and both gene expression data measured with cDNA microarrays and genotype data have missing values. Furthermore, often some of the variables in the data set are continuous and some are discrete and to use standard algorithms for learning a Bayesian network from data, the continuous variables are usually discretized with potential loss of information.

Mixed Variable Networks The process of learning Bayesian networks requires two components: a search procedure to scan through a set of possible models and a scoring metric, such as BIC or the marginal likelihood, to select one of the models. We have shown in Section 2.2 that when the variables in the network are all continuous, a closed-form solution for the calculation of the marginal likelihood exists under the assumption that each variable is normally distributed around a mean, which *linearly* depends on its parent variables [44, 102]. The drawback is that they are heavily limited in their representation power, as they can only capture linear dependencies among continuous variables. To increase their scope, Gaussian linear networks have been extended into mixture of Gaussian networks, which model a conditional distribution as a weighted mixture of linear Gaussian distributions and can, in principle, represent a wider variety of interactions. Unfortunately, no closed form solution exists to compute the marginal likelihood of these distributions, and we have to resort to computationally demanding approximation methods [14]. The normality assumption on the variables can be relaxed to the more general case that the variables have distributions in the exponential family, and we have introduced the family of GGNs to describe dependency structures of non-normal variables with possibly non-linear dependencies. The crucial assumption in GGNs is that all variables in the network have probability distributions in the same family. An important and yet unsolved issue is the learning of mixed networks, in which some variables are continuous and some are discrete. Imposing the assumption that discrete variables can only be parent nodes in the network, but cannot be children of any continuous Gaussian node leads to a closed form solution for the computation of the marginal likelihood [56]. This property has been applied, for example, to model-based clustering by [74], and it is commonly used in classification problems [10]. However, this restriction can quickly become unrealistic and greatly limit the set of models to explore. As a consequence, common practice is still to discretize continuous variables with possible loss of information, particularly when the continuous variables are highly skewed.

Missing Data The received view of the effect of missing data on statistical inference is based on the approach described by Rubin in [76]. This approach classifies the missing data mechanism as ignorable or not, according to whether the data are missing completely at random (MCAR), missing at random (MAR), or informatively missing (IM). According to this approach, data are MCAR if the probability that an entry is missing is independent of both observed and unobserved values. They are MAR if this probability is at most a function of the observed values in the database and, in all other cases, data are IM. The received view is that, when data are either MCAR or MAR, the missing data mechanism is ignorable for parameter estimation, but it is not when data are IM.

An important but overlooked issue is whether the missing data mechanism generating data that are MAR is ignorable for model selection [77, 85]. We have shown that this is not the case for the class of graphical models exemplified in Figure 16 [85]. We assume that there is only one variable with missing data (the variable Y_4 in the DAG) and that its possible parents are all fully observed. To model the missing data mechanism, we introduce the dummy variable R that takes on one of the two values: $R = 1$ when Y_4 is observed, and $R = 0$ when Y_4 is missing. The missing data mechanism can be described by the graphical structure relating R, Y_4 and Y_1, Y_2, Y_3: when R is not linked to any of the variables, data are MCAR; when R is linked to any subset of Y_1, Y_2, Y_3 but not Y_4, data are MAR; when R is linked to Y_4, data are IM. If the

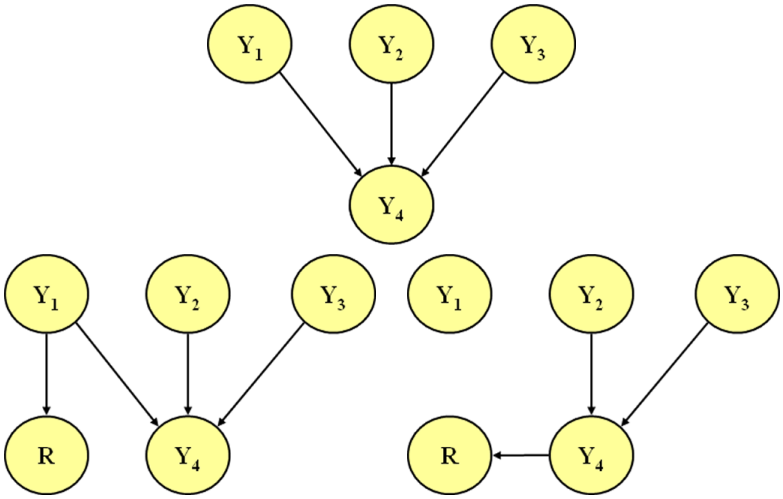


Figure 16: *An example of partially ignorable missing data mechanism. Top: The variable Y_4 in the Bayesian network is only partially observed, while the parents Y_1, Y_2, Y_3 are fully observed. Bottom left: The variable R encodes whether Y is observed ($R = 1$) or not ($R = 0$). Because the variable R is a child of Y_1, which is fully observed, data are MAR. Bottom right: Removing the variable Y_1 from the dependency model for Y_4 induces a link between Y and R so that the missing data mechanism becomes informative.*

graphical structure is known, the missing data mechanism is ignorable for parameter estimation in the first two cases. However, when the task is to learn the graphical structure from data, only a mechanism generating data that are MCAR is ignorable. This fact is shown in the bottom right graph of Figure 16: when we assess the dependency of Y_4 on Y_2, Y_3 but not Y_1, the variable R is linked to Y_4 so that the missing data mechanism is informative for this model structure.

We defined this mechanism only partially ignorable for model selection and we showed how to discriminate between ignorable and partially ignorable missing data mechanisms [85]. We also introduced two approaches to model selection with partially ignorable missing data mechanisms: *ignorable imputation* and *model folding*. Contrary to standard imputation schemes [35, 60, 80, 100, 101], ignorable imputation accounts for the missing-data mechanism and produces, asymptotically, a proper imputation model as defined by Rubin [76, 78]. However, the computation effort can be very demanding and model folding is a deterministic method to approximate the exact marginal likelihood that reaches high accuracy at a low computational cost, because the complexity of the model search is not affected by the presence of incomplete cases. Both ignorable imputation and model folding reconstruct a completion of the incomplete data by taking into account the variables responsible for the missing data. This property is in agreement with the suggestion put forward in [45, 60, 75] that the variables responsible for the missing data should be kept in the model. However, our approach allows us to also evaluate the likelihoods of models that do not depend explicitly on these variables.

Although this work provides the analytical foundations for a proper treatment of missing data when the inference task is model selection, it is limited to the very special situation in which only one variable is partially observed, data are supposed to be only MCAR or MAR, and the set of Bayesian networks is limited to those in which the partially observed variable is a child of the other variables. Research is needed to extend these results to the more general graphical structures, in which several variables can be partially observed and data can be MCAR, MAR or IM.

Acknowledgments

This research was supported by the National Science Foundation (0120309), the Spanish State Office of Education and Universities, the European Social Fund and the Fulbright Program. We thank the reviewers and editors for their invaluable comments that helped improving the original version of this chapter.

References

[1] R. Aebersold and M. Mann. Mass spectrometry-based proteomics. *Nature*, 422(6928):198–207, 2004.

[2] D. Altshuler, J. N. Hirschhorn, M. Klannemark, C. M. Lindgren, M. Vohl, J. Nemesh, C. R. Lane, S. F. Schaffner, S. Bolk, C. Brewer, T. Tuomi, D. Gaudet, T. J. Hudson, M. Daly, L. Groop, and E. S. Lander. The common PPARγ pro12ala polymorphism is associated with decreased risk of type 2 diabetes. *Nat. Genet.*, 26:76–80, 2000.

[3] Amir ben Dor, Laurakay Burhn, Nir Friedman, Iftach Nachman, Michêl Schummer, and Zohar Yakhini. Tissue classification with gene expression profiles. In *The Sixth Annual International Conference on Research in Computational Molecular Biology*, pages 54–64, 2000.

[4] S. G. Bottcher and C. Dethlefsen. Deal: A package for learning Bayesian networks. Available from http://www.jstatsoft.org/v08/i20/deal.pdf.

[5] U. M. Braga-Neto and E. R. Dougerthy. Is cross-validation valid for small-sample microarray classification. *Bioinformatics*, 20:374–380, 2004.

[6] W.W. Cai, J.H. Mao, C.W. Chow, S. Damani, A. Balmain, and A. Bradley. Genome-wide detection of chromosomal imbalances in tumors using bac microarrays. *Nat. Biotechnol.*, 20(4):393–6, 2002.

[7] Z Cai, E.F. Tsung, V. D. Marinescu, M. F. Ramoni, A. Riva, and I. S. Kohane. A Bayesian approach to discovering pathogenic SNPs in conserved protein domains. *Human Mut*, 2004. To appear.

[8] Z. Cai, E.F. Tsung, V.D. Marinescu, M.F. Ramoni, A. Riva, and I.S. Kohane. A Bayesian approach to discovering pathogenic snps in conserved protein domains. *Human Mut.*, 2004. To appear.

[9] E. Castillo, J. M. Gutierrez, and A. S. Hadi. *Expert Systems and Probabilistic Network Models*. Springer, New York, NY, 1997.

[10] P. Cheeseman and J. Stutz. Bayesian classification (AutoClass): Theory and results. In U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, editors, *Advances in Knowledge Discovery and Data Mining*, pages 153–180. MIT Press, Cambridge, MA, 1996.

[11] T. Chen, H.L. He, and G.M. Church. Modeling gene expression with differential equations. *Proceedings of the Pacific Symposium on Biocomputing*, 1999.

[12] J. Cheng and M. Druzdzel. AIS-BN: An adaptive importance sampling algorithm for evidential reasoning in large Bayesian networks. *J. Artif. Intell. Res.*, 13:155–188, 2000.

[13] D. M. Chickering. Learning equivalence classes of Bayesian-network structures. *J. Mach. Learn. Res.*, 2:445–498, February 2002.

[14] D. M. Chickering and D. Heckerman. Efficient approximations for the marginal likelihood of incomplete data. *Mach. Learn.*, 29:181–212, 1997.

[15] F.S. Collins, M.S. Guyer, and A. Chakravarti. Variations on a theme: Cataloging human DNA sequence variation. *Science*, 278:1580–1581, 1997.

[16] G. F. Cooper and E. Herskovitz. A Bayesian method for the induction of probabilistic networks from data. *Mach. Learn.*, 9:309–347, 1992.

[17] R. G. Cowell, A. P. Dawid, S. L. Lauritzen, and D. J. Spiegelhalter. *Probabilistic Networks and Expert Systems*. Springer, New York, NY, 1999.

[18] K. D. Dahlquist, N. Salomonis, K. Vranizan, S. C. Lawlor, and B. R. Conklin. GenMAPP, a new tool for viewing and analyzing microarray data on biological pathways. *Nat. Genet.*, 31:19–20, 2002.

[19] A. P. Dawid. Conditional independence in statistical theory (with discussion). *J. Roy. Statist. Soc. B*, 41:1–31, 1979.

[20] A. P. Dawid. Conditional independence for statistical operation. *Ann. Statist.*, 8:598–617, 1980.

[21] A. P. Dawid and S. L. Lauritzen. Hyper Markov laws in the statistical analysis of decomposable graphical models. *Ann. Statist.*, 21:1272–1317, 1993. Correction ibidem, (1995), *23*, 1864.

[22] M.J.L. de Hoon, S. Imoto, K. Kobayashi, N. Ogasawara, and S. Miyano. Inferring gene regulatory networks from time-ordered gene expression data of Bacillus subtilis using differential equations. In *Proceedings of the Pacific Symposium on Biocomputing*, volume 8, pages 17–28, 2003.

[23] S. M. Dhanasekaran, T. R. Barrette, D. Ghosh, R. Shah, S. Varambally, K. Kurachi, K. J. Pienta, M. A. Rubin, and A. M. Chinnaiyan. Delineation of prognostic biomarkers in prostate cancer. *Nature*, 412:822–826, 2001.

[24] S. W. Doniger, N. Salomonis, K. D. Dahlquist, K. Vranizan, S. C. Lawlor, and B. R. Conklin. MAPPFinder: Using gene ontology and genMAPP to create a global gene-expression profile from microarray data. *Genome Biol.*, 2003. Available from http://genomebiology.com/2003/4/1/R7.

[25] R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis*. Wiley, New York, NY, 1973.

[26] M. Essand, G. Vasmatzis, U. Brinkmann, P. Duray, B. Lee, and I Pastan. High expression of a specific t-cell receptor γ transcript in epithelial cells of the prostate. *Proc. Natl. Acad. Sci. USA*, 96:9287–9292, 1999.

[27] N. Freimer and C. Sabatti. The human phenome project. *Nat. Genet.*, 34(1):15–21, 2003.

[28] N. Friedman. Inferring cellular networks using probabilistic graphical models. *Science*, 303:799–805, 2004.

[29] N. Friedman, D. Geiger, and M. Goldszmidt. Bayesian network classifiers. *Mach. Learn.*, 29:131–163, 1997.

[30] N. Friedman and D. Koller. Being Bayesian about network structure: A Bayesian approach to structure discovery in bayesian networks. *Mach. Learn.*, 50:95–125, 2003.

[31] N. Friedman, M. Linial, I. Nachman, and D. Pe'er. Using bayesian network to analyze expression data. *J. Comput. Biol.*, 7:601–620, 2000.

[32] N. Friedman, K. Murphy, and S. Russell. Learning the structure of dynamic probabilistic networks. In *Proceedings of the 14th Annual Conference on Uncertainty in Artificial Intelligence (UAI-98)*, pages 139–147, San Francisco, CA, 1998. Morgan Kaufmann Publishers.

[33] D. Geiger and D. Heckerman. Learning gaussian networks. In *Proceedings of the Tenth Annual Conference on Uncertainty in Artificial Intelligence (UAI-94)*, San Francisco, 1994. Morgan Kaufmann.

[34] D. Geiger and D. Heckerman. A characterization of Dirichlet distributions through local and global independence. *Ann. Statist.*, 25:1344–1368, 1997.

[35] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin. *Bayesian Data Analysis*. Chapman and Hall, London, UK, 1995.

[36] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Trans. Pattern Anal. and Mach. Intell.*, 6:721–741, 1984.

[37] W. R. Gilks and G. O. Roberts. Strategies for improving MCMC. In W. R. Gilks, S. Richardson, and D. J. Spiegelhalter, editors, *Markov Chain Monte Carlo in Practice*, pages 89–114. Chapman and Hall, London, UK, 1996.

[38] R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, 1 M. L. Loh H. Coller, J. R. Downing, M. A. Caligiuri, C. D. Bloomfield, and E. S. Lander. Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*, 286:531–537, 1999.

[39] I. J. Good. Rational decisions. *J. Roy. Statist. Soc. B*, 14:107–114, 1952.

[40] I. J. Good. *The Estimation of Probability: An Essay on Modern Bayesian Methods*. MIT Press, Cambridge, MA, 1968.

[41] D. J. Hand. *Construction and Assessment of Classification Rules*. Wiley, New York, NY, 1997.

[42] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer-Verlag, New York, 2001.

[43] D. Heckerman. Bayesian networks for data mining. *Data Mining and Knowledge Discovery*, 1:79–119, 1997.

[44] D. Heckerman, D. Geiger, and D. M. Chickering. Learning Bayesian networks: The combinations of knowledge and statistical data. *Mach. Learn.*, 20:197–243, 1995.

[45] D. F. Heitjan and D. B. Rubin. Ignorability and coarse data. *Ann. Statist.*, 19:2244–2253, 1991.

[46] R. Jansen, H. Yu, D. Greenbaum, Y. Kluger, N. J. Krogan, S. Chung, A. Emili, M. Snyder, J. F. Greenblatt, and M. Gerstein. A Bayesian networks approach for predicting protein-protein interactions from genomic data. *Science*, 302:449–453, 2003.

[47] N. P. Jewell. *Statistics for Epidemiology*. CRC/Chapman and Hall, Boca Raton, 2003.

[48] R. E. Kass and A. Raftery. Bayes factors. *J. Amer. Statist. Assoc.*, 90:773–795, 1995.

[49] A. D. Keller, M. Schummer, L. Hood, and W. L. Ruzzo. Bayesian classification of DNA array expression data. Technical Report UW-CSE-2000-08-01, Department of Computer Science and Engineering, Seattle, WA, 2000.

[50] B. Krishnapuram, L. Carin, and A. Hartemink. Joint classifier and feature optimization for comprehensive cancer diagnosis using gene expression data. *J. Comput. Biol.*, to appear, 2004.

[51] E. S. Lander. Array of hope. *Nat. Genet.*, 21:3–4, 1999. Supplement.

[52] E.S. Lander. The new genomics: Global views of biology. *Science*, 274:536–539, 1996.

[53] P. Langley, W. Iba, and K. Thompson. An analysis of Bayesian classifiers. In *Proceedings of the Tenth National Conference on Artificial Intelligence*, pages 223–228, Menlo Park, CA, 1992. AAAI Press.

[54] P. Langley and S. Sage. Induction of selective bayesian classifiers. In *Proceedings of the Tenth Conference on Uncertainty in Artificial Intelligence*, pages 399–406, Seattle, WA, USA, 1994. Morgan Kaufmann.

[55] P. Larranaga, C. Kuijpers, R. Murga, and Y. Yurramendi. Learning Bayesian network structures by searching for the best ordering with genetic algorithms. *IEEE Trans. Pattern Anal. and Mach. Intell.*, 26:487–493, 1996.

[56] S. L. Lauritzen. Propagation of probabilities, means and variances in mixed graphical association models. *J. Amer. Statist. Assoc.*, 87(420):1098–108, 1992.

[57] S. L. Lauritzen. *Graphical Models*. Oxford University Press, Oxford, UK, 1996.

[58] S. L. Lauritzen and D. J. Spiegelhalter. Local computations with probabilities on graphical structures and their application to expert systems (with discussion). *J. Roy. Statist. Soc. B*, 50:157–224, 1988.

[59] Y. Li, C. Campbell, and M. Tipping. Bayesian automatic relevance determination algorithms for classifying gene expression data. *Bioinformatics*, 18:1332–1339, 2002.

[60] R. J. A. Little and D. B. Rubin. *Statistical Analysis with Missing Data*. Wiley, New York, NY, 1987.

[61] D.J. Lockhart, H. Dong, M.C. Byrne, M.T. Follettie, M.V. Gallo, M.S. Chee, M. Mittmann, C. Wang, M. Kobayashi, H. Horton, and E.L. Brown. Expression monitoring by hybridization to high-density oligonucleotide arrays. *Nat. Biotechnol.*, 14:1675–1680, 1996.

[62] D. Madigan and A. E. Raftery. Model selection and accounting for model uncertainty in graphical models using Occam's window. *J. Amer. Statist. Assoc.*, 89:1535–1546, 1994.

[63] D. Madigan and G. Ridgeway. Bayesian data analysis for data mining. In *Handbook of Data Mining*, pages 103–132. MIT Press, 2003.

[64] D. Madigan and J. York. Bayesian graphical models for discrete data. *Int. Statist. Review*, pages 215–232, 1995.

[65] P. McCullagh and J. A. Nelder. *Generalized Linear Models*. Chapman and Hall, London, 2nd edition, 1989.

[66] A. O'Hagan. *Bayesian Inference*. Kendall's Advanced Theory of Statistics. Arnold, London, UK, 1994.

[67] J. Ott. *Analysis of Human Genetic Linkage*. Johns Hopkins University Press, Baltimore, 1999.

[68] G. Parmigiani, E. S. Garrett, R. Anbazhagan, and E. Gabrielson. A statistical framework for expressionbased molecular classification in cancer. *J. Roy. Statist. Soc. B*, 64:717–736, 2002. (With Discussion).

[69] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of plausible inference*. Morgan Kaufmann, San Francisco, CA, 1988.

[70] D. Peér, A. Regev, G. Elidan, and N. Friedman. Inferring subnetworks to analyze expression data. *Bioinformatics*, 17:215–224, 2001.

[71] E. Phizicky, P.I.H. Bastiaens, H. Zhu, M. Snyder, and S. Fields. Protein analysis on a proteomic scale. *Nature*, 422(6928):208–215, 2004.

[72] M. Ramoni, A. Riva, M. Stefanelli, and V. Patel. An ignorant belief network to forecast glucose concentration from clinical databases. *Artif. Intell. Med.*, 7:541–559, 1995.

[73] M. Ramoni and P. Sebastiani. Bayesian methods. In *Intelligent Data Analysis. An Introduction*, pages 131–168. Springer, New York, NY, 2nd edition, 2003.

[74] M. Ramoni, P. Sebastiani, and I.S. Kohane. Cluster analysis of gene expression dynamics. *Proc. Natl. Acad. Sci. USA*, 99(14):9121–6, 2002.

[75] D. B. Rubin. Inference and missing data. *Biometrika*, 63:581–592, 1976.

[76] D. B. Rubin. *Multiple Imputation for Nonresponse in Survey*. Wiley, New York, NY, 1987.

[77] D. B. Rubin. Multiple imputation after 18 years. *J. Amer. Statist. Assoc.*, 91:473–489, 1996.

[78] D. B. Rubin, H. S. Stern, and V. Vehovar. Handling "don't know" survey responses: the case of the Slovenian plebiscite. *J. Amer. Statist. Assoc.*, 90:822–828, 1995.

[79] M. Sahami. Learning limited dependence Bayesian classifiers. In *Proceeding of the 2 Int. Conf. On Knowledge Discovery & Data Mining*, 1996.

[80] J. L. Schafer. *Analysis of Incomplete Multivariate Data*. Chapman and Hall, London, UK, 1997.

[81] M. Schena, D. Shalon, R. W. Davis, and P. O. Brown. Quantitative monitoring of gene expression patterns with a complementary DNA microarray. *Science*, 270:467–70, 1995.

[82] J. M. Schildkraut. Examining complex genetic interactions. In *Gene Mapping in Complex Human Diseases*, pages 379–410. John Wiley & Sons, New York, 1998.

[83] P. Sebastiani, E. Gussoni, I. S. Kohane, and M. Ramoni. Statistical challenges in functional genomics (with discussion). *Statist. Sci.*, 18:33–70, 2003.

[84] P. Sebastiani, J. Jeneralczuk, and Marco Ramoni. Screening experiments with microarrays. In A. Dean and S Lewis, editors, *Screening*. Springer, 2004. To appear.

[85] P. Sebastiani and M. Ramoni. Bayesian selection of decomposable models with incomplete data. *J. Amer. Statist. Assoc.*, 96(456):1375–1386, 2001.

[86] P. Sebastiani and M. Ramoni. Generalized gamma networks. Technical report, University of Massachusetts, Department of Mathematics and Statistics, 2003.

[87] P. Sebastiani, M. Ramoni, and A. Crea. Profiling customers from in-house data. *ACM SIGKDD Explorations*, 1:91–96, 2000.

[88] P. Sebastiani, M. Ramoni, and I. Kohane. BADGE: Technical notes. Technical report, Department of Mathematics and Statistics, University of Massachusetts at Amherst, 2003.

[89] P. Sebastiani and M. F. Ramoni. On the use of Bayesian networks to analyze survey data. *Res. Offic. Statist.*, 4:54–64, 2001.

[90] P. Sebastiani, M. F. Ramoni, V. Nolan, C. Baldwin, and M. H. Steinberg. Discovery of complex traits associated with overt stroke in patients with sickle cell anemia by Bayesian network modeling. In *27th Annual Meeting of the National Sickle Cell Disease Program*, 2004. To appear.

[91] P. Sebastiani, M. F. Ramoni, V. Nolan, C. T. Baldwin, and M. H. Steinberg. Genetic dissection and prognostic modeling of overt stroke in sickle cell anemia. 2004. Submitted.

[92] E. Segal, M. Shapira, A. Regev, D. Pe'er, D. Botstein, D. Koller, and N. Friedman. Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data. *Nat. Genet.*, 34(2):166–76, 2003.

[93] E. Segal, B. Taskar, A. Gasch, N. Friedman, and Daphne Koller. Rich probabilistic models for gene expression. *Bioinformatics*, 1:1–9, 2001.

[94] I. Shmulevich, E. R. Dougherty, K. Seungchan, and W. Zhang. Probabilistic boolean networks: A rule based uncertainty model for gene regulatory networks. *Bioinformatics*, 18:261–274, 2002.

[95] D. Singh, P. G. Febbo, K. Ross, D. G. Jackson, J. Manola, C. Ladd, P. Tamayo, A. A. Renshaw, A. V. DAmico, J. P. Richie, E. S. Lander, M. Loda, P. W. Kantoff, T. R. Golub, and W. R. Sellers. Gene expression correlates of clinical prostate cancer behavior. *Cancer Cell*, 1:203–209, 2002.

[96] M. Singh and M. Valtorta. Construction of Bayesian network structures from data: A brief survey and an efficient algorithm. *Intern. J. Approx. Reason.*, 12:111–131, 1995.

[97] D. J. Spiegelhalter and S. L. Lauritzen. Sequential updating of conditional probabilities on directed graphical structures. *Networks*, 20:157–224, 1990.

[98] P. Spirtes, C. Glymour, and R. Scheines. *Causation, prediction and search*. Springer, New York, 1993.

[99] P. Spirtes, C. Glymour, and R. Scheines. Constructing bayesian network models of gene expression networks from microarray data. In *Proceedings of the Atlantic Symposium on Computational Biology, Genome Information Systems & Technology*, 2000.

[100] M. A. Tanner. *Tools for Statistical Inference*. Springer, New York, NY, third edition, 1996.

[101] Y. Thibaudeau and W. E. Winler. Bayesian networks representations, generalized imputation, and synthetic microdata satisfying analytic restraints. Technical report, Statistical Research Division report RR 2002/09, 2002. http://www.census.gov/srd/www/byyear.html.

[102] B. Thiesson. Accelerated quantification of Bayesian networks with incomplete data. In *Proceedings of the First ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 306–311, New York, NY, 1995. ACM Press.

[103] A. Thomas, D. J. Spiegelhalter, and W. R. Gilks. Bugs: A program to perform Bayesian inference using Gibbs Sampling. In J. Bernardo, J. Berger, A. P. Dawid, and A. F. M. Smith, editors, *Bayesian Statistics 4*, pages 837–42. Oxford University Press, Oxford, UK, 1992.

[104] M. West, C. Blanchette, H. Dressman, E. Huang, S. Ishida, R. Spang, H. Zuzan, J. A. Olson Jr, J. R. Marks, and J. R. Nevins. Predicting the clinical status of human breast cancer by using gene expression profiles. *Proc. Natl. Acad. Sci. USA*, 98:11462–11467, 2001.

[105] J. Whittaker. *Graphical Models in Applied Multivariate Statistics*. Wiley, New York, NY, 1990.

[106] C. Yoo, V. Thorsson, and G.F. Cooper. Discovery of causal relationships in a gene-regulation pathway from a mixture of experimental and observational DNA microarray data. In *Proceedings of the Pacific Symposium on Biocomputing*, 2002. Available from http://psb.stanford.edu.

[107] J. Yu, V. Smith, P. Wang, A. Hartemink, and E. Jarvis. Using Bayesian network inference algorithms to recover molecular genetic regulatory networks. In *International Conference on Systems Biology 2002 (ICSB02)*, 2002.

[108] H. Zhou and S. Sakane. Sensor planning for mobile robot localization using Bayesian network inference. *J. of Advanced Robotics*, 16, 2002. To appear.

[109] Y. Zhu, J. Hollmén, R. Raty, Y. Aalto, B. Nagy, E. Elonen, J. Kere, H. Mannila, K. Franssila, and S. Knuutila. Investigatory and analytical approaches to differential gene expression profiling in mantle cell lymphoma. *British J. Haematol.*, 119:905–905, 2002.

Index