Chapter 2

Probability, Statistics, and Computational Science

Niko Beerenwinkel and Juliane Siebourg

Abstract

In this chapter, we review basic concepts from probability theory and computational statistics that are fundamental to evolutionary genomics. We provide a very basic introduction to statistical modeling and discuss general principles, including maximum likelihood and Bayesian inference. Markov chains, hidden Markov models, and Bayesian network models are introduced in more detail as they occur frequently and in many variations in genomics applications. In particular, we discuss efficient inference algorithms and methods for learning these models from partially observed data. Several simple examples are given throughout the text, some of which provide the basis for models that are discussed in more detail in subsequent chapters.

Key words Bayesian inference, Bayesian networks, Dynamic programming, EM algorithm, Hidden Markov models, Markov chains, Maximum likelihood, Statistical models

1 Statistical Models

Evolutionary genomics can only be approached with the help of statistical modeling. Stochastic fluctuations are inherent to many biological systems. Specifically, the evolutionary process itself is stochastic, with random mutations and random mating being major sources of variation. In general, stochastic effects play an increasingly important role if the number of molecules, or cells, or individuals of a population is small. Stochastic variation also arises from measurement errors. Biological data is often noisy due to experimental limitations, especially for high-throughput technologies, such as microarrays or next-generation sequencing [1, 2].

Statistical modeling addresses the following questions: What can be generalized from a finite sample obtained from an experiment to the population? What can be learned about the underlying biological mechanisms? How certain can we be about our model predictions?

In the frequentist view of statistics, the observed variability in the data is the result of a fixed true value being perturbed by

Maria Anisimova (ed.), *Evolutionary Genomics: Statistical and Computational Methods*, Methods in Molecular Biology, vol. 1910, https://doi.org/10.1007/978-1-4939-9074-0_2,

random variation, such as, for example, measurement noise. Probabilities are thus interpreted as long-run expected relative frequencies. By contrast, from a Bayesian point of view, probabilities represent our uncertainty about the state of nature. There is no true value, but only the data is real. Our prior belief about an event is updated in light of the data.

Statistical models represent the observed variability or uncertainty by probability distributions [3, 4]. The observed data are regarded as realizations of random variables. The parameters of a statistical model are usually the quantities of interest because they describe the amount and nature of systematic variation in the data. Parameter estimation and model selection are discussed in more detail in the next section. In this section, we first consider discrete, and then continuous random variables and univariate (1-dimensional) before multivariate (n-dimensional) ones. We start by formulating the well-known Hardy–Weinberg principle [5, 6] as a statistical model.

Example 1 (Hardy–Weinberg Model): The Hardy–Weinberg model is a statistical model for the genotypes in a diploid population of infinite size. Let us assume that there are two alleles, denoted A and a, and hence three genotypes, denoted AA, Aa = aA, and aa. Let X be the random variable with state space $\mathcal{X} = \{\text{AA}, \text{Aa}, \text{aa}\}$ describing the genotype. We parametrize the probability distribution of X by the allele frequency p of A and the allele frequency $q = 1 - p$ of a. The Hardy–Weinberg model is defined by:

$$P(X = \text{AA}) = p^2, \tag{1}$$

$$P(X = \text{Aa}) = 2p(1-p), \tag{2}$$

$$P(X = \text{aa}) = (1-p)^2. \tag{3}$$

The parameter space of the model is $\Theta = \{p \in \mathbb{R} \mid 0 \le p \le 1\} = [0,1]$, the unit interval. We denote the Hardy–Weinberg model by $\text{HW}(p)$ and write $X \sim \text{HW}(p)$ if X follows the distribution (Eqs. 1–3). □

The Hardy–Weinberg distribution $P(X)$ is a discrete probability distribution (or probability mass function) with finite state space: We have $0 \le P(X = x) \le 1$ for all $x \in \mathcal{X}$ and $\sum_{x \in \mathcal{X}} P(X = x) = p^2 + 2p(1-p) + (1-p)^2 = [p + (1-p)]^2 = 1$. In general, any statistical model for a discrete random variable with n states defines a subset of the $(n-1)$-dimensional probability simplex:

$$\Delta_{n-1} = \left\{(p_1, \ldots, p_n) \in [0,1]^n \mid p_1 + \cdots + p_n = 1\right\}. \tag{4}$$

The probability simplex is the set of all possible probability distributions of X, and statistical models can be understood as specific subsets of the simplex [7].

The Hardy–Weinberg distribution is of interest because it arises under the assumption of random mating. A population with major allele frequency p has genotype probabilities given in Eqs. 1–3 after one round of random mating. We find that the new allele frequency:

$$p' = P(\mathrm{AA}) + P(\mathrm{Aa})/2 = p^2 + 2p(1-p)/2 = p, \tag{5}$$

is equal to the one in the previous generation. Thus, genetic variation is preserved under this simple model of sexual reproduction, and the population is at equilibrium after one generation. In other words, Eqs. 1–3 describe the set of all populations at Hardy–Weinberg equilibrium. The parametric representation:

$$\left\{(p_{\mathrm{AA}},p_{\mathrm{Aa}},p_{\mathrm{aa}}) \in \Delta_2 \mid p_{\mathrm{AA}} = p^2,\ p_{\mathrm{Aa}} = 2p(1-p),\ p_{\mathrm{aa}} = (1-p)^2\right\}, \tag{6}$$

of this set of distributions is equivalent to the implicit representation as the intersection of the Hardy–Weinberg curve:

$$4\ p_{\mathrm{AA}}\ p_{\mathrm{aa}} - p_{\mathrm{Aa}}^2 = 0 \tag{7}$$

with the probability simplex Δ_2 (Fig. 1).

The simplest discrete random variable is a binary (or Bernoulli) random variable X. The textbook example of a Bernoulli trial is the flipping of a coin. The state space of this random experiment is the set that contains all possible outcomes, namely, whether the coin lands on heads ($X = 0$) or tails ($X = 1$). We write $\mathcal{X} = \{0,1\}$ to denote this state space. The parameter space is the set that contains all possible values of the model parameters. In the coin tossing example, the only parameter is the probability of observing tails, p, and this parameter can take any value between 0 and 1, so we write $\Theta = \{p \mid 0 \leq p \leq 1\}$ for the parameter space. In general, the event $X = 1$ is often called a "success," and $p = P(X = 1)$ the probability of success.

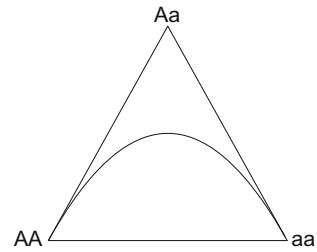


Fig. 1 De Finetti diagram showing the Hardy–Weinberg curve $4\ p_{\mathrm{AA}}\ p_{\mathrm{aa}} - p_{\mathrm{Aa}}^2 = 0$ inside the probability simplex $\Delta_2 = \{(p_{\mathrm{AA}},\ p_{\mathrm{Aa}},\ p_{\mathrm{aa}}) | p_{\mathrm{AA}} + p_{\mathrm{Aa}} + p_{\mathrm{aa}} = 1\}$. Each point in this space represents a population as described by its genotype frequencies. Points on the curve correspond to populations in Hardy–Weinberg equilibrium

Example 2 (Binomial Distribution): Consider n independent Bernoulli trials, each with success probability p. Let X be the random variable counting the number of successes k among the n trials. Then, X has state space $\mathcal{X} = \{0, \ldots, n\}$ and

$$P(X = k) = \binom{n}{k} p^k (1-p)^{n-k}. \tag{8}$$

This is the binomial distribution, denoted $\mathrm{Binom}(n, p)$. Its parameter space is $\Theta = \mathbb{N} \times [0,1]$. Examples of binomially distributed random variables are the number of "heads" in n successive coin tosses or the number of mutated genes in a group of species. □

Important characteristics of a probability distribution are its expectation (or expected value, or mean) and its variance. They are defined, respectively, as:

$$\mathrm{E}(X) = \sum_{x \in \mathcal{X}} x \; P(X = x), \tag{9}$$

$$\mathrm{Var}(X) = \sum_{x \in \mathcal{X}} [x - \mathrm{E}(X)]^2 \; P(X = x). \tag{10}$$

The standard deviation is $\sqrt{\mathrm{Var}(X)}$. For the binomial distribution, $X \sim \mathrm{Binom}(n, p)$, we find $\mathrm{E}(X) = np$ and $\mathrm{Var}(X) = np(1-p)$.

Example 3 (Poisson Distribution): The Poisson distribution $\mathrm{Pois}(\lambda)$ with parameter $\lambda \geq 0$ is defined as:

$$P(X = k) \quad = \quad \frac{\lambda^k \; e^{-\lambda}}{k!}, \quad k \in \mathbb{N}. \tag{11}$$

It describes the number X of independent events occurring in a fixed period of time (or space) at average rate λ and independently of the time since (or distance to) the last event. The Poisson distribution has equal expectation and variance, $\mathrm{E}(X) = \mathrm{Var}(X) = \lambda$. □

The Poisson distribution is used frequently as a model for the number of DNA mutations in a gene after a certain time period, where λ is the mutation rate. Both the binomial and the Poisson distribution describe counts of random events. In the limit of large n and fixed product np, the two distributions coincide, $\mathrm{Binom}(n, p) \to \mathrm{Pois}(np)$, for $n \to \infty$.

Example 4 (Shotgun Sequencing): Let us consider a simplified model of the shotgun approach to DNA sequencing. Suppose that n reads of length L have been obtained from a genome of size G. We assume that all reads have the same probability of being sequenced. Then, the probability of hitting a specific base with one read is $p = L/G$, and the average coverage of the sequencing run is $c = np$. Under this model, the number of times X a single base is sequenced is distributed as $\mathrm{Binom}(n, p)$. For large n, we have

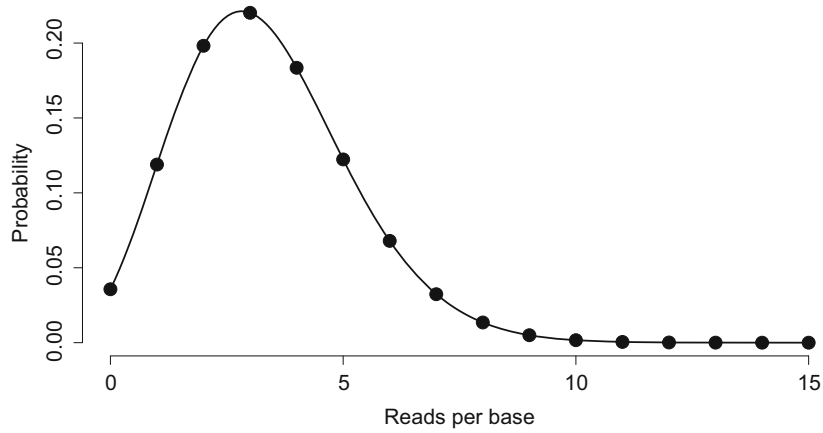


Fig. 2 Coverage distribution of a shotgun sequencing experiment with $n = 10^8$ reads of length $L = 100$ of the human genome of length $G = 3 \cdot 10^9$. The average coverage is $c = np = 3.4$, where $p = L/G$. Dots show the binomial coverage distribution $\text{Binom}(n, p)$ and the solid line its approximation by the Poisson distribution $\text{Pois}(np)$. Note that the Poisson distribution is also discrete and just shown as a line to distinguish it from the binomial distribution

$$P(X = k) = \binom{n}{k} p^k (1-p)^{n-k} \approx \frac{c^k \ e^{-c}}{k!}. \tag{12}$$

For example, using next-generation sequencing technology, one might obtain $n = 10^8$ reads of length $L = 100$ bases in a single run. For the human genome of length $G = 3 \cdot 10^9$, we obtain a coverage of $c = 3.4$. The distribution of the number of reads per base pair is shown in Fig. 2. In particular, the fraction of unsequenced positions is $P(X = 0) = e^{-c} = 3.57\%$. □

A continuous random variable X takes values in $\mathcal{X} = \mathbb{R}$ and is defined by a nonnegative function $f(x)$ such that:

$$P(X \in B) = \int_B f(x)dx, \quad \text{for all subsets } B \subseteq \mathbb{R}. \tag{13}$$

The function f is called the probability density function of X. For an interval:

$$P(X \in [a,b]) = P(a \le X \le b) = \int_a^b f(x)dx. \tag{14}$$

The cumulative distribution function is

$$F(b) = P(X \le b) = \int_{-\infty}^b f(x)dx, \quad b \in \mathbb{R}. \tag{15}$$

Thus, the density is the derivative of the cumulative distribution function, $\frac{d}{dx} F(x) = f(x)$.

In analogy to the discrete case, expectation and variance of a continuous random variable are defined, respectively, as:

$$\mathrm{E}(X) = \int_{-\infty}^{\infty} x\, f(x)\, dx, \tag{16}$$

$$\mathrm{Var}(X) = \int_{-\infty}^{\infty} [x - \mathrm{E}(X)]^2\, f(x)\, dx. \tag{17}$$

Example 5 (Normal Distribution): The normal (or Gaussian) distribution has the density function:

$$f(x) = (2\pi\sigma^2)^{-1/2} \exp\left[-\frac{(x-\mu)^2}{2\sigma^2}\right]. \tag{18}$$

The parameter space is $\Theta = \{(\mu, \sigma^2) \mid \mu \in \mathbb{R},\ \sigma^2 \in \mathbb{R}_+\}$. A normal random variable $X \sim \mathrm{Norm}(\mu, \sigma^2)$ has mean $\mathrm{E}(X) = \mu$ and variance $\mathrm{Var}(X) = \sigma^2$. $\mathrm{Norm}(0,1)$ is called the standard normal distribution. □

The normal distribution is frequently used as a model for measurement noise. For example, $X \sim \mathrm{Norm}(\mu, \sigma^2)$ might describe the hybridization intensity of a sample to a probe on a microarray. Then, μ is the level of expression of the corresponding gene and σ^2 summarizes the experimental noise associated with the microarray experiment. The parameters can be estimated from a finite sample $\{x^{(1)}, \ldots, x^{(N)}\}$, i.e., from N replicate experiments, as the empirical mean and variance, respectively:

$$\overline{x} = \frac{1}{N} \sum_{i=1}^{N} x^{(i)}, \tag{19}$$

$$s^2 = \frac{1}{N-1} \sum_{i=1}^{N} (x^{(i)} - \overline{x})^2. \tag{20}$$

The normal distribution plays a special role in statistics due to the central limit theorem. It asserts that the average $\bar{X}_N = (X^{(1)} + \cdots + X^{(N)})/N$ of N independent (see below) and identically distributed (i.i.d.) random variables $X^{(i)}$ with equal mean μ and variance σ^2 converges in distribution to the standard normal distribution:

$$\sqrt{N}\left(\frac{\bar{X}_N - \mu}{\sigma}\right) \xrightarrow{d} \mathrm{Norm}(0,1), \tag{21}$$

irrespective of the shape of their distribution. As a consequence, many test statistics and estimators are asymptotically normally distributed. For example, the Poisson distribution $\mathrm{Pois}(\lambda)$ is approximately normal $\mathrm{Norm}(\lambda, \lambda)$ for large values of λ.

We often measure multiple quantities at the same time, for example the expression of several genes, and are interested in correlations among the variables. Let X and Y be two random

variables with expected values μ_X and μ_Υ and variances σ_X^2 and σ_Υ^2, respectively. The covariance between X and Υ is

$$\mathrm{Cov}(X, \Upsilon) = \mathrm{E}[(X - \mu_X)(\Upsilon - \mu_\Upsilon)] = \mathrm{E}[X\Upsilon] - \mathrm{E}[X]\mathrm{E}[\Upsilon] \tag{22}$$

and the correlation between X and Υ is $\rho_{X,\Upsilon} = \mathrm{Cov}(X, \Upsilon)/(\sigma_X \sigma_\Upsilon)$. For observations $(x^{(1)}, y^{(1)}), \ldots, (x^{(N)}, y^{(N)})$, the sample correlation coefficient is

$$r_{x,y} = \frac{\sum_{i=1}^{N} (x^{(i)} - \overline{x})(y^{(i)} - \overline{y})}{(N-1) s_X s_\Upsilon}, \tag{23}$$

where s_X and s_Υ are the sample standard deviations of X and Υ, respectively, defined in Eq. 20.

So far, we have worked with univariate distributions and we now turn to multivariate distributions, i.e., we consider random vectors $X = (X_1, \ldots, X_n)$ such that each X_i is a random variable. For the case of discrete random variables X_i, we first generalize the binomial distribution to random experiments with a finite number of outcomes.

Example 6 (Multinomial Distribution): Let K be the number of possible outcomes of a random experiment and θ_k the probability of outcome k. We consider the random vector $X = (X_1, \ldots, X_K)$ with values in $\mathcal{X} = \mathbb{N}^K$, where X_k counts the number of outcomes of type k. The multinomial distribution $\mathrm{Mult}(n, \theta_1, \ldots, \theta_K)$ is defined as:

$$P(X = x) = \frac{n!}{x_1! \cdots x_K!} \theta_1^{x_1} \cdots \theta_K^{x_K} \tag{24}$$

if $\sum_{k=1}^{K} x_k = n$, and 0 otherwise. The parameter space of the model is $\Theta = \mathbb{N} \times \Delta_{K-1}$. For $K = 2$, we recover the binomial distribution (Eq. 8). Each component X_k of a multinomial vector has expected value $\mathrm{E}(X_k) = n\theta_k$ and $\mathrm{Var}(X_k) = n\theta_k(1 - \theta_k)$. The covariance of two components is $\mathrm{Cov}(X_k, X_l) = -n\theta_k\theta_l$, for $k \neq l$. □

In general, the covariance matrix Σ of a random vector X is defined by:

$$\Sigma_{ij} = \mathrm{Cov}(X_i, X_j) = \mathrm{E}[(X_i - \mu_i)(X_j - \mu_j)], \tag{25}$$

where μ_i is the expected value of X_i. The matrix Σ is also called the variance–covariance matrix because the diagonal terms are the variances $\Sigma_{ii} = \mathrm{Cov}(X_i, X_i) = \mathrm{Var}(X_i)$.

A continuous multivariate random variable X takes values in $\mathcal{X} = \mathbb{R}^n$. It is defined by its cumulative distribution function:

$$F(x) = P(X \leq x), \quad x \in \mathbb{R}^n \tag{26}$$

or, equivalently, by the probability density function:

$$f(x) = \frac{\partial^n}{\partial x_1 \cdots \partial x_n} F(x_1, \ldots, x_n), \quad x \in \mathbb{R}^n. \tag{27}$$

Example 7 (Multivariate Normal Distribution): For $n \geq 1$ and $x \in \mathbb{R}^n$, the multivariate normal (or Gaussian) distribution has density:

$$f(x) = (2\pi)^{-n/2} \det(\Sigma)^{-1/2} \exp\left[-\frac{1}{2}(x-\mu)^t \Sigma^{-1}(x-\mu)\right], \tag{28}$$

with parameter space $\Theta = \{(\mu, \Sigma) \mid \mu = (\mu_1, \ldots, \mu_n) \in \mathbb{R}^n$ and $\Sigma = \left(\sigma_{ij}^2\right) \in \mathbb{R}^{n \times n}\}$, where Σ is the symmetric, positive-definite covariance matrix and μ the expectation. We write $X = (X_1, \ldots, X_n) \sim \text{Norm}(\mu, \Sigma)$ for a random vector with such a distribution. □

We say that two random variables X and Υ are independent if $P(X, \Upsilon) = P(X)P(\Upsilon)$ or, equivalently, if the conditional probability $P(X \mid \Upsilon) = P(X, \Upsilon)/P(\Upsilon)$ is equal to the unconditional probability $P(X)$. If X and Υ are independent, denoted $X \perp \Upsilon$, then $\text{E}[X\Upsilon] = \text{E}[X]\text{E}[\Upsilon]$ and $\text{Var}(X + \Upsilon) = \text{Var}(X) + \text{Var}(\Upsilon)$. It follows that independent random variables have covariance zero. However, the converse is only true in specific situations, for example if (X, Υ) is multivariate normal, but not in general because correlation captures only linear dependencies.

This limitation can be addressed by using statistical models which allow for a richer dependency structure. Subheading 7 is devoted to Bayesian networks, a family of probabilistic graphical models based on conditional independences. Let X, Υ, and Z be three random vectors. Generalizing the notion of statistical independence, we say that X is conditionally independent of Υ given Z and write $X \perp \Upsilon \mid Z$ if $P(X, \Upsilon \mid Z) = P(X \mid Z)P(\Upsilon \mid Z)$. Bayes' theorem states that

$$P(\Upsilon \mid X) = \frac{P(X \mid \Upsilon)P(\Upsilon)}{P(X)}, \tag{29}$$

where $P(\Upsilon)$ is called the prior probability and $P(\Upsilon \mid X)$ the posterior probability. Intuitively, the prior $P(\Upsilon)$ encodes our a priori knowledge about Υ (i.e., before observing X), and $P(\Upsilon \mid X)$ is our updated knowledge about Υ a posteriori (i.e., after observing X).

We have $P(X) = \Sigma_\Upsilon P(X, \Upsilon)$ if Υ is discrete, and similarly $P(X) = \int_\Upsilon P(X, \Upsilon) d\Upsilon$ if Υ is continuous. Here, $P(X)$ is called the marginal and $P(X, \Upsilon)$ the joint probability. This summation or integration is known as marginalization (Fig. 3).

Since $P(X) = \Sigma_\Upsilon P(X, \Upsilon) = \Sigma_\Upsilon P(X \mid \Upsilon)P(\Upsilon)$, Bayes' theorem can also be rewritten as:

$$P(\Upsilon \mid X) = \frac{P(X \mid \Upsilon)P(\Upsilon)}{\sum_{y' \in \mathcal{Y}} P(X \mid y')P(y')}, \tag{30}$$

where $P(y') = P(\Upsilon = y')$ and $\mathcal{Y}$ is the state space of Υ.

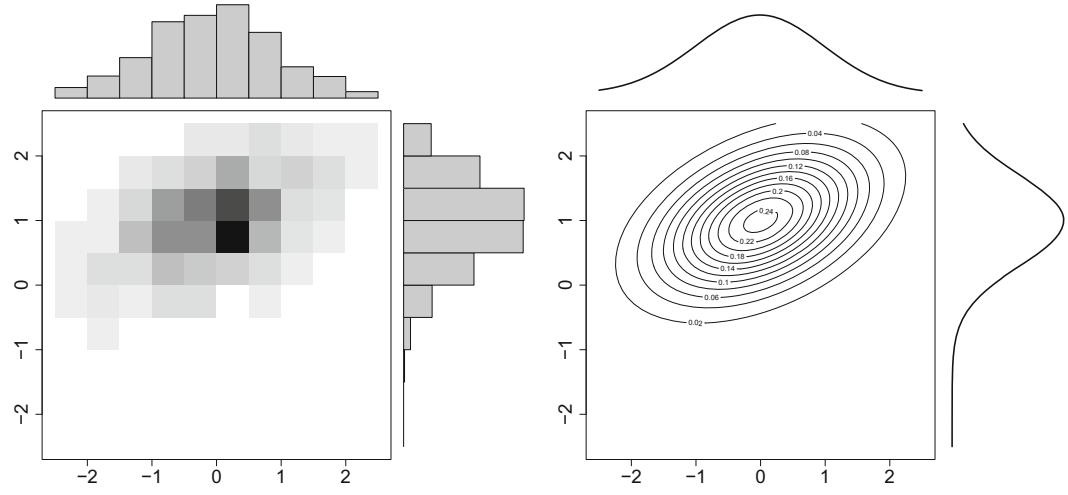


Fig. 3 Marginalization. Left: two-dimensional histogram of a discrete bivariate distribution with the two marginal histograms. Right: contour plot of a two-dimensional Gaussian density with the marginal distributions of each component

Example 8 (Diagnostic Test): We want to evaluate a diagnostic test for a rare genetic disease. The binary random variables D and T indicate disease status ($D = 1$, diseased) and test result ($T = 1$, positive), respectively. Let us assume that the prevalence of the disease is 0.5%, i.e., 0.5% of all people in the population are known to be affected. The test has a false positive rate (probability that somebody is tested positive who does not have the disease) of $P(T = 1 \mid D = 0) = 5\%$ and a true positive rate (probability that somebody is tested positive who has the disease) of $P(T = 1 \mid D = 1) = 90\%$. Then, the posterior probability of a person having the disease given that he or she tested positive is

$$P(D = 1 \mid T = 1) = \frac{P(T = 1 \mid D = 1)P(D = 1)}{P(T = 1 \mid D = 0)P(D = 0) + P(T = 1 \mid D = 1)P(D = 1)} = 0.083, \tag{31}$$

that is, only 8.3% of the positively tested individuals actually have the disease. Thus, our prior belief of the disease status, $P(D)$, has been modified in light of the test result by multiplication with $P(T \mid D)$ to obtain the updated belief $P(D \mid T)$. □

Exercise 9 (Conditional Independence): Let X, Υ, and Z be random variables. Using the laws of probability, show that X and Υ are conditionally independent given Z (i.e., $X \perp \Upsilon \mid Z$) if and only if $P(X \mid \Upsilon, Z) = P(X \mid Z)$.

2 Statistical Inference

Statistical models have parameters and a common task is to estimate the model parameters from observed data. The goal is to find the set of parameters with the best model fit. There are two major approaches to parameter estimation: maximum likelihood (ML) and Bayes.

The maximum likelihood approach is based on the likelihood function. Let us consider a fixed statistical model M with parameter space Θ and assume that we have observed realizations $\mathcal{D} = \{x^{(1)}, \ldots, x^{(N)}\}$ of the discrete random variable $X \sim \mathrm{M}(\theta_0)$ for some unknown parameter $\theta_0 \in \Theta$. For the fixed data set $\mathcal{D}$, the likelihood function of the model is

$$L(\theta) = P(\mathcal{D} \mid \theta), \tag{32}$$

where we write $P(\mathcal{D} \mid \theta)$ to emphasize that, here, the probability of the data depends on the model parameter θ. For continuous random variables, the likelihood function is defined similarly in terms of the density function, $L(\theta) = f(\mathcal{D} \mid \theta)$. Maximum likelihood estimation seeks the parameter $\theta \in \Theta$ for which $L(\theta)$ is maximal. Rather than $L(\theta)$, it is often more convenient to maximize $\ell(\theta) = \log L(\theta)$, the log-likelihood function. If the data are i.i.d., then:

$$\ell(\theta) = \sum_{i=1}^{N} \log P(X = x^{(i)} \mid \theta). \tag{33}$$

Example 10 (Likelihood Function of the Binomial Model): Suppose we have observed $k = 7$ successes in a total of $N = 10$ Bernoulli trials. The likelihood function of the binomial model (Eq. 8) is

$$L(p) = p^k \, (1 - p)^{N-k}, \tag{34}$$

where p is the success probability (Fig. 4). To maximize L, we consider the log-likelihood function:

$$\ell(p) = \log L(p) = k \log(p) + (N - k) \log(1 - p) \tag{35}$$

and the likelihood equation $d\ell/dp = 0$. The ML estimate (MLE) is the solution $\hat{p}_{\mathrm{ML}} = k/N = 7/10$. Thus, the MLE of the success probability is just the relative frequency of successes. □

Example 11 (Likelihood Function of the Hardy–Weinberg Model): If we genotype a finite random sample of a population of diploid individuals at a single locus, then the resulting data consists of the numbers of individuals n_{AA}, n_{Aa}, and n_{aa} with the respective genotypes. Assuming Hardy–Weinberg equilibrium (Eqs. 1–3), we want to estimate the allele frequencies p and $q = 1 - p$ of the

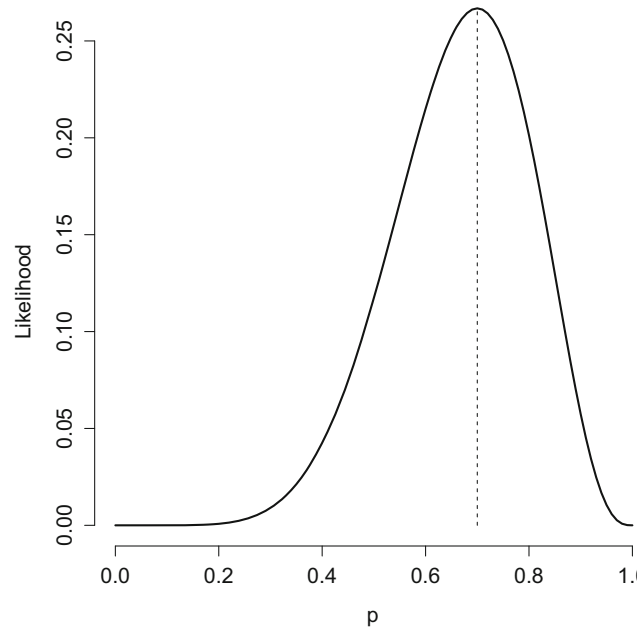


Fig. 4 Likelihood function of the binomial model. The underlying data set consists of $k = 7$ successes out of $N = 10$ Bernoulli trials. The likelihood $L(p) = p^k (1 - p)^{N-k}$ is plotted as a function of the model parameter p, the probability of success (solid line). The MLE is the maximum of this function, $\hat{p}_{\mathrm{ML}} = k/N = 7/10$ (dashed line)

population. The likelihood function of the Hardy–Weinberg model is $L(p) = P(\mathrm{AA})^{n_{\mathrm{AA}}} P(\mathrm{Aa})^{n_{\mathrm{Aa}}} P(\mathrm{aa})^{n_{\mathrm{aa}}}$ and the log-likelihood is

$$\begin{aligned} \ell(p) &= n_{\mathrm{AA}} \log p^2 + n_{\mathrm{Aa}} \log 2p(1-p) + n_{\mathrm{aa}} \log(1-p)^2 \\ &\propto (2n_{\mathrm{AA}} + n_{\mathrm{Aa}}) \log p + (n_{\mathrm{Aa}} + 2n_{\mathrm{aa}}) \log(1-p), \end{aligned} \tag{36}$$

where we have dropped the constant $n_{\mathrm{Aa}} \log 2$. The MLE of $p \in [0, 1]$ can be found by maximizing ℓ. Solving the likelihood equation:

$$\frac{\partial \ell}{\partial p} = \frac{2n_{\mathrm{AA}} + n_{\mathrm{Aa}}}{p} - \frac{n_{\mathrm{Aa}} + 2n_{\mathrm{aa}}}{1-p} = 0 \tag{37}$$

yields the MLE $\hat{p}_{\mathrm{ML}} = (2n_{\mathrm{AA}} + n_{\mathrm{Aa}})/(2N)$, where $N = n_{\mathrm{AA}} + n_{\mathrm{Aa}} + n_{\mathrm{aa}}$ is the total sample size. For example, if we sample $N = 100$ genotypes with $n_{\mathrm{AA}} = 81$, $n_{\mathrm{Aa}} = 18$, and $n_{\mathrm{aa}} = 1$, then we find $\hat{p}_{\mathrm{ML}} = (2 \cdot (81 + 18))/(2 \cdot 100) = 0.9$ for the frequency of the major allele. □

MLEs have many desirable properties. Asymptotically, as the sample size $N \to \infty$, they are normally distributed, unbiased, and have minimal variance. The uncertainty in parameter estimation associated with the sampling variance of the finite data set can be quantified in confidence intervals. There are several ways to

construct confidence intervals and statistical tests for MLEs based on the asymptotic behavior of the log-likelihood function $\ell(\theta) = \log L(\theta)$ and its derivatives. For example, the asymptotic normal distribution of the MLE is

$$\hat{\theta}_{\mathrm{ML}} \overset{a}{\sim} \mathrm{Norm}\left(\theta, J(\theta)^{-1}\right), \tag{38}$$

where $I(\theta) = -\partial^2\ell/\partial\theta^2$ is the Fisher information and $J(\theta) = \mathrm{E}[I(\theta)]$ the expected Fisher information. This result gives rise to the Wald confidence intervals:

$$\left[\hat{\theta}_{\mathrm{ML}} \pm z_{1-\alpha/2}\, J(\theta)^{-1}\right], \tag{39}$$

where $z_{1-\alpha/2} = \inf\{x \in \mathbb{R} \mid 1-\alpha/2 \le F(x)\}$ is the $(1-\alpha/2)$ quantile and F the cumulative distribution function of the standard normal distribution. Equation 38 still holds after replacing $J(\theta)$ with the standard error $\mathrm{se}(\hat{\theta}_{\mathrm{ML}}) = [I(\hat{\theta}_{\mathrm{ML}})]^{-\frac{1}{2}}$ or $[J(\hat{\theta}_{\mathrm{ML}})]^{-\frac{1}{2}}$, and it also generalizes to higher dimensions. Other common constructions of confidence intervals include those based on the asymptotic distribution of the score function $S(\theta) = \partial\ell/\partial\theta$ and the log-likelihood ratio $\log(L(\hat{\theta}_{\mathrm{ML}})/L(\theta))$ [8].

We now discuss another more generic approach to quantify parameter uncertainty, not restricted to ML estimation, which is applied frequently in practice due to its simple implementation. Bootstrapping [9] is a resampling method in which independent observations are resampled from the data with replacement. The resulting new data set consists of (some of) the original observations, and under i.i.d. assumptions, the bootstrap replicates have asymptotically the same distribution as the data. Intuitively, by sampling with replacement, one is pretending that the collection of replicates thus obtained is a good proxy for the distribution of data sets that one would have obtained, had we been able to actually replicate the experiment. In this way, the variability of an estimator (or more generally the distribution of any test statistic) can be approximated by evaluating the estimator (or the statistic) on a collection of bootstrap replicates. For example, the distribution of the ML estimator of a model parameter θ can be obtained from the bootstrap samples.

Example 12 (Bootstrap Confidence Interval for the ML Allele Frequency): We use bootstrapping to estimate the distribution of the ML estimator $\hat{p}_{\mathrm{ML}}$ of the Hardy–Weinberg model for the data set $(n_{\mathrm{AA}}, n_{\mathrm{Aa}}, n_{\mathrm{aa}}) = (81, 18, 1)$ of Example 11. For each bootstrap sample, we draw $N = 100$ genotypes with replacement from the original data to obtain random integer vectors of length three summing to 100. The ML estimate is computed for each of a total of B bootstrap samples. The resulting distributions of $\hat{p}_{\mathrm{ML}}$ are shown in Fig. 5, for $B = 100$, 1000, and 10,000. The means of these empirical distributions are 0.899, 0.9004, and 0.9001,

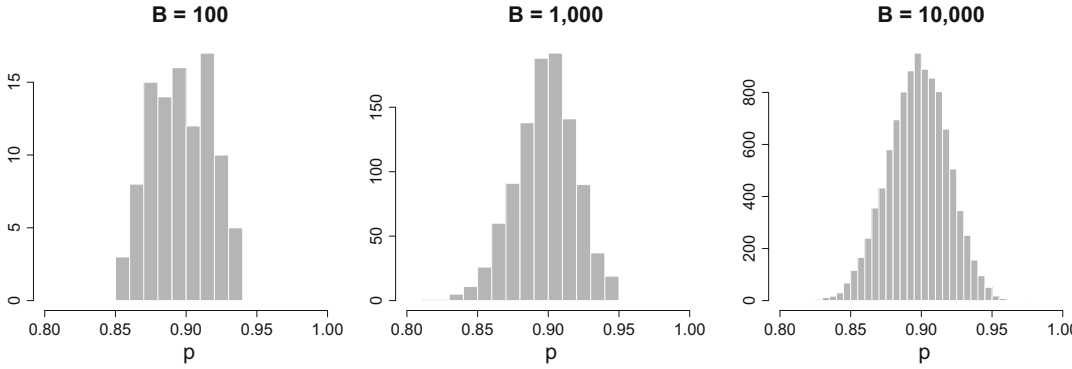


Fig. 5 Bootstrap analysis of the ML allele frequency. The bootstrap distribution of the maximum likelihood estimator $\hat{p}_{\mathrm{ML}} = (2n_{\mathrm{AA}} + n_{\mathrm{Aa}})/(2N)$ of the major allele frequency in the Hardy–Weinberg model is plotted for $B = 100$ (left), $B = 1000$ (center), and $B = 10,000$ (right) bootstrap samples, for the data set (n_{AA}, n_{Aa}, n_{aa}) = (81, 18, 1)

respectively, and 95% bootstrap confidence intervals can be derived from the 2.5 and 97.5% quantiles of the distributions. For $B = 100$, 1000, and 10,000, we obtain, respectively, [0.8598, 0.9350], [0.860, 0.940], and [0.855, 0.940]. The basic bootstrap confidence intervals have several limitations, including bias of the bootstrap estimator and skewness of the bootstrap distribution. Other methods exist for constructing confidence intervals from the bootstrap distribution to address some of them [9]. □

The Bayesian approach takes a different point of view and regards the model parameters as random variables [10]. Inference is then concerned with estimating the joint distribution of the parameters θ given the observed data $\mathcal{D}$. By Bayes' theorem (Eq. 30), we have

$$P(\theta \mid \mathcal{D}) = \frac{P(\mathcal{D} \mid \theta)P(\theta)}{P(\mathcal{D})} = \frac{P(\mathcal{D} \mid \theta)P(\theta)}{\int_{\theta \in \Theta} P(\mathcal{D} \mid \theta)P(\theta)\, d\theta}, \tag{40}$$

that is, the posterior probability of the parameters is proportional to the likelihood of the data times the prior probability of the parameters. It follows that, for a uniform prior, the mode of the posterior is equal to the MLE.

From the posterior, credible intervals of parameter estimates can be derived such that the parameter lies in the interval with a certain probability, say 95%. This is in contrast to a 95% confidence interval in the frequentist approach because, there, the parameter is fixed and the interval boundaries are random variables. The meaning of a confidence interval is that 95% of similar intervals would contain the true parameter, if intervals were constructed independently from additional identically distributed data.

The prior $P(\theta)$ encodes our a priori belief in θ before observing the data. It can be used to incorporate domain-specific knowledge

into the model, but it may also be uninformative or objective, in which case all observations are equally likely, or nearly so, a priori. However, it can sometimes be difficult to find noninformative priors. In practice, conjugate priors are most often used. A conjugate prior is one that is invariant with respect to the distribution family under multiplication with the likelihood, i.e., the posterior belongs to the same family as the prior. Conjugate priors are mathematically convenient and computationally efficient because the posterior can be calculated analytically for a wide range of statistical models.

Example 13 (Dirichlet Prior): Let $T = (T_1, \ldots, T_K)$ be a continuous random variable with state space Δ_{K-1}. The Dirichlet distribution $\mathrm{Dir}(\alpha)$ with parameters $\alpha \in \mathbb{R}_+^K$ has probability density function:

$$f(\theta_1, \ldots, \theta_K) = \frac{\Gamma\left(\sum_{i=1}^K \alpha_i\right)}{\prod_{i=1}^K \Gamma(\alpha_i)} \prod_{i=1}^K \theta_i^{\alpha_i - 1}, \tag{41}$$

where Γ is the gamma function. The Dirichlet prior is conjugate to the multinomial likelihood: If $T \sim \mathrm{Dir}(\alpha)$ and $(X \mid T = \theta) \sim \mathrm{Mult}(n, \theta_1, \ldots, \theta_K)$, then $(\theta \mid X = x) \sim \mathrm{Dir}(\alpha + x)$. For $K = 2$, this distribution is called the beta distribution. Hence, the beta distribution is the conjugate prior to the binomial likelihood. □

Example 14 (Posterior Probability of Genotype Frequencies): Let us consider the simple genetic system with two loci and two alleles each of Example 1, but without assuming the Hardy–Weinberg model. We regard the observed genotype frequencies $(n_{\mathrm{AA}}, n_{\mathrm{Aa}}, n_{\mathrm{aa}}) = (81, 18, 1)$ as the result of a draw from a multinomial distribution $\mathrm{Mult}(n, \theta_{\mathrm{AA}}, \theta_{\mathrm{Aa}}, \theta_{\mathrm{aa}})$. Assuming a Dirichlet prior $\mathrm{Dir}(\alpha_{\mathrm{AA}}, \alpha_{\mathrm{Aa}}, \alpha_{\mathrm{aa}})$, the posterior genotype probabilities follow the Dirichlet distribution $\mathrm{Dir}(\alpha_{\mathrm{AA}} + n_{\mathrm{AA}}, \alpha_{\mathrm{Aa}} + n_{\mathrm{Aa}}, \alpha_{\mathrm{aa}} + n_{\mathrm{aa}})$. In Fig. 6, the prior $\mathrm{Dir}(10, 10, 10)$ is shown on the left, the multinomial likelihood $P((n_{\mathrm{AA}}, n_{\mathrm{Aa}}, n_{\mathrm{aa}}) = (81, 18, 1) \mid \theta_{\mathrm{AA}}, \theta_{\mathrm{Aa}}, \theta_{\mathrm{aa}})$ in the center, and the resulting posterior $\mathrm{Dir}(10 + 81, 10 + 18, 10 + 1)$ on the right. Note that the MLE is different from the mode of the posterior. As compared to the likelihood, the nonuniform prior has shifted the maximum of the posterior toward the center of the probability simplex. □

We often have two or more competing models and would like to assess which one describes best the given data. For example, we may have observed genotypes from the set {AA, Aa, aa} and want to test whether the Hardy–Weinberg model (Example 1) is a more appropriate description of the genotype data than the multinomial model of the previous Example 14. Intuitively, we might want to select the model that fits the data best, for example, by comparing

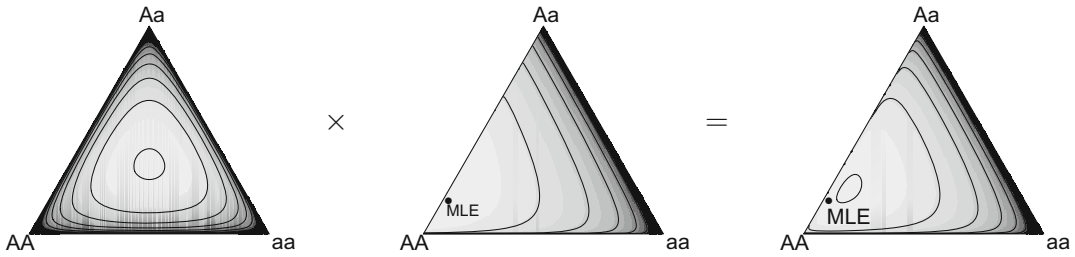


Fig. 6 Dirichlet prior for multinomial likelihood. The Dirichlet prior is conjugate to the multinomial likelihood. Shown are contour lines of the prior $\mathrm{Dir}(10, 10, 10)$ on the left, the multinomial likelihood $P((n_{\mathrm{AA}}, n_{\mathrm{Aa}}, n_{\mathrm{aa}}) = (81, 18, 1)|\theta_{\mathrm{AA}}, \theta_{\mathrm{Aa}}, \theta_{\mathrm{aa}})$ in the center, and the resulting posterior $\mathrm{Dir}(91, 28, 11)$ on the right. The posterior is the product of prior and likelihood

their likelihoods. However, the Hardy–Weinberg model has only one parameter, namely the allele frequency p, whereas the multinomial model has three parameters subject to the constraint $\theta_{\mathrm{AA}} + \theta_{\mathrm{Aa}} + \theta_{\mathrm{aa}} = 1$. Hence, the number of free parameters is one and two, respectively, for the two models. This difference in the complexity of the models makes a comparison based only on the goodness of fit invalid, because models with more parameters, i.e., higher complexity, can generally provide a better fit. Estimating model complexity and scoring models based on both model complexity and goodness of fit is therefore essential for model comparison and model selection.

The goal of model selection is to find the model that best generalizes to unseen data, rather than just fits the observed data, because we seek the model capable of the most accurate predictions. A model that fits well but generalizes poorly is said to overfit the data. Models that are too complex tend to overfit the data. Model selection can be regarded as finding the right level model complexity for the given data, such that the predictive performance is optimized. This involves defining a criterion of optimality and a procedure for finding the optimal model.

A common frequentist approach to model selection are likelihood ratios. For a data set $\mathcal{D}$, we compare a null model, M_0, to an alternative model, M_1, at given point estimates using the ratio of their likelihoods:

$$\Lambda(\mathcal{D}) = \frac{L(\hat{\theta}_0)}{L(\hat{\theta}_1)} \tag{42}$$

If $\Lambda(\mathcal{D}) < c$, for a defined threshold c, we reject the null model and favor the alternative model. The choice of c should be informed by the distribution of Λ under the null. If the two models are nested, i.e., if M_0 can be obtained from M_1 by specifying a subset of the parameters, then $-2\log\Lambda$ is approximately χ^2-distributed with degrees of freedom equal to the difference in the number of free parameters between M_1 and M_0.

In the Bayesian framework, it is natural to compare the posterior probabilities of the two models. By Bayes theorem, we have, for $i = 0, 1$:

$$P(\mathrm{M}_i \mid \mathcal{D}) = \frac{P(\mathcal{D} \mid \mathrm{M}_i)P(\mathrm{M}_i)}{P(\mathcal{D})} \tag{43}$$

where:

$$P(\mathcal{D} \mid \mathrm{M}_i) = \int P(\mathcal{D} \mid \theta_i,\ \mathrm{M}_i)P(\theta_i \mid \mathrm{M}_i)\ d\theta_i \tag{44}$$

is the marginal likelihood. The marginal likelihood accounts for model complexity and for uncertainty in parameter estimates, but is usually analytically intractable and costly to compute. Various approximations of the marginal likelihood exist that give rise to model selection scores, such as the Bayesian information criterion (BIC; *see* Subheading 7) and the Akaike information criterion (AIC) [11].

For Bayesian model comparison, we consider the posterior odds:

$$\frac{P(\mathrm{M}_0 \mid \mathcal{D})}{P(\mathrm{M}_1 \mid \mathcal{D})} = \frac{P(\mathcal{D} \mid \mathrm{M}_0)}{P(\mathcal{D} \mid \mathrm{M}_1)}\ \frac{P(\mathrm{M}_0)}{P(\mathrm{M}_1)} \tag{45}$$

The ratio of the marginal likelihoods, i.e., the first factor on the right-hand side of Eq. 45, is called the Bayes factor. With equal priors, a Bayes factor larger than 20 is often considered strong support for M_0 over M_1 [12].

Exercise 15 (Poisson Distribution): We wish to model the number of bacterial colonies in a Petri dish and assume that the count data of this experiment follows a Poisson distribution Pois(λ) (Example 3). Derive the log-likelihood function of this model and calculate the MLE of the model parameter λ. Suppose now that the number of bacterial colonies on a Petri dish follows the Poisson distribution with mean $\lambda = 5$. What is the probability of finding exactly three colonies?

3 Hidden Data and the EM Algorithm

We often cannot observe all relevant random variables due to, for example, experimental limitations or study designs. In this case, a statistical model $P(X, Z \mid \theta \in \Theta)$ consists of the observed random variable X and the hidden (or latent) random variable Z, both of which can be multivariate. In this section, we write $X = (X^{(1)}, \ldots, X^{(N)})$ for the random variables describing the N observations and refer to X also as the observed data. The hidden data for this model is $Z = (Z^{(1)}, \ldots, Z^{(N)})$ and the complete data is (X, Z). For

convenience, we assume the parameter space Θ to be continuous and the state spaces $\mathcal{X}$ of X and $\mathcal{Z}$ of Z to be discrete.

In the Bayesian framework, one does not distinguish between unknown parameters and hidden data, and it is natural to assess the joint posterior $P(\theta, Z \mid X) \propto P(X \mid \theta, Z)P(\theta, Z)$, which is $P(X, Z \mid \theta)P(\theta)$ if priors are independent, i.e., if $P(\theta, Z) = P(\theta)P(Z)$. Alternatively, if the distribution of the hidden data Z is not of interest, it can be marginalized out. Then, the posterior (Eq. 40) becomes

$$P(\theta \mid X) = \frac{\sum_Z P(X, Z \mid \theta)P(\theta)}{\int_{\theta\in\Theta}\sum_Z P(X, Z \mid \theta)P(\theta)\, d\theta}. \tag{46}$$

In the likelihood framework, it can be more efficient to estimate the hidden data, rather than marginalizing over it. The hidden (or complete-data) log-likelihood is

$$\ell_{\text{hid}}(\theta) = \log P(X, Z \mid \theta) = \sum_{i=1}^{N} \log P(X^{(i)}, Z^{(i)} \mid \theta). \tag{47}$$

For ML parameter estimation, we need to consider the observed log-likelihood:

$$\begin{aligned} \ell_{\text{obs}}(\theta) &= \log P(X \mid \theta) = \log \sum_Z P(X, Z \mid \theta) \\ &= \log \sum_{Z^{(1)}\in\mathcal{Z}} \cdots \sum_{Z^{(N)}\in\mathcal{Z}} \prod_{i=1}^{N} P(X^{(i)}, Z^{(i)} \mid \theta). \end{aligned} \tag{48}$$

This likelihood function is usually very difficult to maximize and one has to resort to numerical optimization techniques. Generic local methods, such as gradient descent or Newton's method, can be used, but there is also a more specific local optimization procedure, which avoids computing any derivatives of the likelihood function, called the expectation maximization (EM) algorithm [13].

In order to maximize the likelihood function (Eq. 48), we consider any distribution $q(Z)$ of the hidden data Z and write

$$\ell_{\text{obs}}(\theta) = \log \sum_Z q(Z) \frac{P(X, Z \mid \theta)}{q(Z)} = \log \mathrm{E}[P(X, Z \mid \theta)/q(Z)], \tag{49}$$

where the expected value is with respect to $q(Z)$. Jensen's inequality applied to the concave log function asserts that $\log \mathrm{E}[\Upsilon] \geq \mathrm{E}[\log \Upsilon]$. Hence, the observed log-likelihood is bounded from below by $\mathrm{E}[\log(P(X, Z \mid \theta)/q(Z))]$, or

$$\ell_{\text{obs}}(\theta) \geq \mathrm{E}[\ell_{\text{hid}}(\theta)] + H(q), \tag{50}$$

where $H(q) = -\mathrm{E}[\log q(Z)]$ is the entropy. The idea of the EM algorithm is to maximize this lower bound instead of $\ell_{\mathrm{obs}}(\theta)$ itself. Intuitively, this task is easier because the big sum over the hidden data in Eq. 48 disappears on the right-hand side of Eq. 50 upon taking expectations.

The EM algorithm is an iterative procedure alternating between an E step and an M step. In the E step, the lower bound (Eq. 50) is maximized with respect to the distribution q by setting $q(Z) = P(Z \mid X, \theta^{(t)})$, where $\theta^{(t)}$ is the current estimate of θ, and computing the expected value of the hidden log-likelihood:

$$Q(\theta \mid \theta^{(t)}) = \mathrm{E}_{Z|X,\theta^{(t)}}[\ell_{\mathrm{hid}}(\theta)]. \tag{51}$$

In the M step, Q is maximized with respect to θ to obtain an improved estimate:

$$\theta^{(t+1)} = \arg\max_{\theta} Q(\theta \mid \theta^{(t)}). \tag{52}$$

The sequence $\theta^{(1)}, \theta^{(2)}, \theta^{(3)}, \ldots$ converges to a local maximum of the likelihood surface (Eq. 48). The global maximum and, hence, the MLE is generally not guaranteed to be found with this local optimization method. In practice, the EM algorithm is often run repeatedly with many different starting solutions $\theta^{(1)}$, or with few very reasonable starting solutions obtained from other heuristics or educated guesses.

Example 16 (Naive Bayes): Let us assume that we observe realizations of a discrete random variable $(X_1, \ldots, X_L)$ and we want to cluster observations into K distinct groups. For this purpose, we introduce a hidden random variable Z with state space $\mathcal{Z} = [K] = \{1, \ldots, K\}$ indicating class membership. The joint probability of $(X_1, \ldots, X_L)$ and Z is

$$\begin{aligned} P(X_1, \ldots, X_L, Z) &= P(Z)P(X_1, \ldots, X_L \mid Z) \\ &= P(Z)\prod_{n=1}^{L} P(X_n \mid Z). \end{aligned} \tag{53}$$

The marginalization of this model with respect to the hidden data Z is the unsupervised naive Bayes model. The observed variables X_n are often called features and Z the latent class variable (Fig. 7).

The model parameters are the class prior $P(Z)$, which we assume to be constant and will ignore, and the conditional probabilities $\theta_{n,kx} = P(X_n = x \mid Z = k)$. The complete-data likelihood of observed data $X = (X^{(1)}, \ldots, X^{(N)})$ and hidden data $Z = (Z^{(1)}, \ldots, Z^{(N)})$ is

$$P(X, Z \mid \theta) = \prod_{i=1}^{N} P(X^{(i)}, Z^{(i)} \mid \theta) = \prod_{i=1}^{N} P(Z^{(i)}) \prod_{n=1}^{L} P(X_n^{(i)} \mid Z^{(i)}) \tag{54}$$

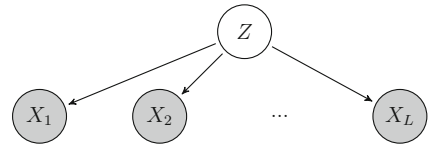


Fig. 7 Graphical representation of the naive Bayes model. Observed features X_n are conditionally independent given the latent class variable Z

$$\propto \prod_{i=1}^{N}\prod_{n=1}^{L}\theta_{n,Z^{(i)}X_n^{(i)}} = \prod_{i=1}^{N}\prod_{n=1}^{L}\prod_{k\in[K]}\prod_{x\in\mathcal{X}}\theta_{n,kx}^{I_{n,kx}(Z^{(i)})}, \tag{55}$$

where $I_{n,kx}(Z^{(i)})$ is equal to one if and only if $Z^{(i)} = k$ and $X_n^{(i)} = x$, and zero otherwise.

To apply the EM algorithm for estimating θ without observing Z, we consider the hidden log-likelihood:

$$\ell_{\text{hid}}(\theta) = \log P(X,Z \mid \theta) = \sum_{i=1}^{N}\sum_{n=1}^{L}\sum_{k\in[K]}\sum_{x\in\mathcal{X}} I_{n,kx}(Z^{(i)})\log\theta_{n,kx}. \tag{56}$$

In the E step, we compute the expected values of $Z^{(i)}$:

$$\begin{aligned}\gamma_{n,kx}^{(i)} = \mathrm{E}_{Z\mid X_n=x,\theta'}\left[Z^{(i)}\right] &= \frac{P(X_n^{(i)} = x \mid Z^{(i)} = k)}{\sum_{k'\in K} P(X_n^{(i)} = x \mid Z^{(i)} = k')} \\ &= \frac{\theta'_{n,kx}}{\sum_{k'\in K}\theta'_{n,k'x}},\end{aligned} \tag{57}$$

where θ' is the current estimate of θ. The expected value $\gamma_{n,kx}^{(i)}$ is sometimes referred to as the responsibility of class k for observation $X_n^{(i)} = x$. The expected hidden log-likelihood can be written in terms of the expected counts $N_{n,kx} = \sum_{i=1}^{N}\gamma_{n,kx}^{(i)}$ as:

$$\mathrm{E}_{Z\mid X,\theta'}[\ell_{\text{hid}}(\theta)] = \sum_{n=1}^{L}\sum_{k\in[K]}\sum_{x\in\mathcal{X}} N_{n,kx}\log\theta_{n,kx}. \tag{58}$$

In the M step, maximization of this sum yields $\hat{\theta}_{n,kx} = N_{n,kx}/\sum_{x'} N_{n,kx'}$. □

4 Markov Chains

A stochastic process $\{X_t, t\in\mathcal{T}\}$ is a collection of random variables with common state space $\mathcal{X}$. The index set $\mathcal{T}$ is usually interpreted as time and X_t is the state of the process at time t. A discrete-time stochastic process $X = (X_1, X_2, X_3, \ldots)$ is called a Markov chain

[14], if $X_{n+1} \perp X_{n-1} \mid X_n$ for all $n \geq 2$ or, equivalently, if each state depends only on its immediate predecessor:

$$P(X_n \mid X_{n-1}, \ldots, X_1) = P(X_n \mid X_{n-1}), \quad \text{for all } n \geq 2. \tag{59}$$

We consider here Markov chains with finite state space $\mathcal{X} = [K] = \{1, \ldots, K\}$ that are homogeneous, i.e., with transition probabilities independent of time:

$$T_{kl} = P(X_{n+1} = l \mid X_n = k), \quad \text{for all } k,l \in [K],\ n \geq 2. \tag{60}$$

The finite-state homogeneous Markov chain is a statistical model denoted $\mathrm{MC}(\Pi, T)$ and defined by the initial state distribution $\Pi \in \Delta_{K-1}$, where $\Pi_k = P(X_1 = k)$, and the stochastic $K \times K$ transition matrix $T = (T_{kl})$.

We can generalize the one-step transition probabilities T_{kl} to:

$$T^n_{kl} = P(X_{n+j} = l \mid X_j = k), \tag{61}$$

the probability of jumping from state k to state l in n time steps. Any $(n + m)$-step transition can be regarded as an n-step transition followed by an m-step transition. Because the intermediate state i is unknown, summing over all possible values yields the decomposition:

$$T^{n+m}_{kl} = \sum_{i=1}^{K} T^n_{ki} T^m_{il}, \quad \text{for all } n,m \geq 1,\ k,l \in [K], \tag{62}$$

known as the Chapman–Kolmogorov equations. In matrix notation, they can be written as $T^{(n+m)} = T^{(n)} T^{(m)}$. It follows that the n-step transition matrix is the n-th matrix power of the one-step transition matrix, $T^{(n)} = T^n$.

A state l of a Markov chain is accessible from state k if $T^n_{kl} > 0$. We say that k and l communicate with each other and write $k \sim l$ if they are accessible from one another. State communication is reflexive ($k \sim k$), symmetric ($k \sim l \Rightarrow l \sim k$), and, by the Chapman–Kolmogorov equations, transitive ($j \sim k \sim l \Rightarrow j \sim l$). Hence, it defines an equivalence relation on the state space. The Markov chain is irreducible if it has a single communication class, i.e., if any state is accessible from any other state.

A state is recurrent if the Markov chain will reenter it with probability one. Otherwise, the state is transient. In finite-state Markov chains, recurrent states are also positive recurrent, i.e., the expected time to return to the state is finite. A state is aperiodic if the process can return to it after any time $n \geq 1$. Recurrence, positive recurrence, and aperiodicity are class properties: if they hold for a state k, then they also hold for all states communicating with k.

A Markov chain is ergodic if it is irreducible, aperiodic, and positive recurrent. An ergodic Markov chain has a unique stationary distribution π given by:

$$\pi_l = \lim_{n\to\infty} T_{kl}^n = \sum_{k=1}^{K} \pi_k T_{kl}, \qquad l \in [K], \quad \sum_{l=1}^{K} \pi_l = 1 \tag{63}$$

independent of the initial distribution Π. In matrix notation, π is the solution of $\pi^t = \pi^t T$.

Example 17 (Two-State Markov Chain): Consider the Markov chain with state space $\{1, 2\}$ and transition probabilities $T_{12} = \alpha > 0$ and $T_{21} = \beta > 0$. Clearly, the chain is ergodic and its stationary distribution π is given by:

$$(\pi_1 \quad \pi_2) = (\pi_1 \quad \pi_2) \begin{pmatrix} 1-\alpha & \alpha \\ \beta & 1-\beta \end{pmatrix} \tag{64}$$

or, equivalently, $\alpha\pi_1 = \beta\pi_2$. With $\pi_1 + \pi_2 = 1$, we obtain $\pi^t = (\alpha + \beta)^{-1}(\alpha, \beta)$. □

In Example 17, if $\alpha = 0$, then state 1 is called an absorbing state because once entered it is never left. In evolutionary biology and population genetics, Markov chains are often used to model evolving populations, and the fixation probability of an allele can be computed as the absorption probability in such models.

Example 18 (Wright–Fisher Process): We consider two alleles, A and a, in a diploid population of size N. The total number of A alleles in generation n is described by a Markov chain X_n with state space $\{0, 1, 2, \ldots, 2N\}$. We assume that individuals mate randomly and that maternal and paternal alleles are chosen randomly such that $(X_{n+1} \mid X_n) \sim \text{Binom}(2N, k/(2N))$, where k is the number of A alleles in generation n. The Markov chain has transition probabilities:

$$T_{kl} = \binom{2N}{l} \left(\frac{k}{2N}\right)^l \left(\frac{2N-k}{2N}\right)^{2N-l}. \tag{65}$$

If the initial number of A alleles is $X_1 = k$, then $\text{E}(X_1) = k$. After binomial sampling, $\text{E}(X_2) = 2N(k/(2N)) = k$ and hence $\text{E}(X_n) = k$ for all $n \geq 0$. The Markov chain has the two absorbing states 0 and $2N$, which correspond, respectively, to extinction and fixation of the A allele. To compute the fixation probability h_k of A given k initial copies of it:

$$h_k = \lim_{n\to\infty} P(X_n = 2N \mid X_1 = k), \tag{66}$$

we consider the expected value, which is equal to k, in the limit as $n \to \infty$ to obtain

$$k = \lim_{n\to\infty} \mathrm{E}(X_n) = 0 \cdot (1 - h_k) + 2N \cdot h_k. \tag{67}$$

Thus, the fixation probability is just $h_k = k/(2N)$, the initial relative frequency of the allele. The Wright–Fisher process [15, 16] is a basic stochastic model for random genetic drift, i.e., for the variation in allele frequencies only due to random sampling. □

If we observe data $X = (X^{(1)}, \ldots, X^{(N)})$ from a finite Markov chain $\mathrm{MC}(\Pi, T)$ of length L, then the likelihood is

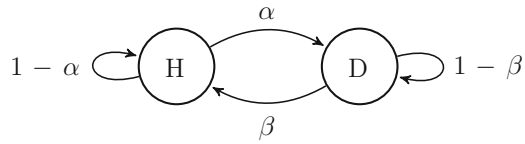
$$\begin{aligned} L(\Pi, T) &= \prod_{i=1}^{N} P(X^{(i)}) = \prod_{i=1}^{N} P(X_1^{(i)}) \prod_{n=1}^{L-1} P(X_{n+1}^{(i)} \mid X_n^{(i)}) \\ &= \prod_{i=1}^{N} \Pi_{X_1^{(i)}} \prod_{n=1}^{L-1} T_{X_n^{(i)}, X_{n+1}^{(i)}}, \end{aligned} \tag{68}$$

which can be rewritten as:

$$\begin{aligned} L(\Pi, T) &= \prod_{i=1}^{N} \prod_{k\in[K]} \Pi_k^{N_k(X^{(i)})} \prod_{k\in[K]} \prod_{l\in[K]} T_{kl}^{N_{kl}(X^{(i)})} \\ &= \prod_{k\in[K]} \Pi_k^{N_k} \prod_{k\in[K]} \prod_{l\in[K]} T_{kl}^{N_{kl}}. \end{aligned} \tag{69}$$

with $N_{kl}(X^{(i)})$ the number of observed transitions from state k into state l in observation $X^{(i)}$, and $N_{kl} = \sum_{i=1}^{N} N_{kl}(X^{(i)})$ the total number of k-to-l transitions in the data, and similarly $N_k(X^{(i)})$ and N_k the number of times the i-th chain, respectively all chains, started in state k.

Exercise 19 (Markov Chains): Let us consider a simple infectious disease model, where each individual is either healthy (H) or diseased (D). We assume the following two-state Markov chain to describe infection-related disease and recovery via clearance of the pathogen:



The probability of a healthy individual becoming sick due to infection is $\alpha = 0.6$, and the probability of a diseased individual to clear the infection and recover is $\beta = 0.9$. The initial probabilities for health and disease are $P(\mathrm{H}) = 0.7$ and $P(\mathrm{D}) = 0.3$. Write down the transition matrix T of this Markov chain. What is the probability of observing the disease trajectories DDHHD and HDHDH? Calculate the stationary distribution of the Markov chain.

5 Continuous-Time Markov Chains

A continuous-time stochastic process $\{X(t), t \geq 0\}$ with finite state space $[K]$ is a continuous-time Markov chain if

$$\begin{aligned} P[X(t+s) = l \mid X(s) = k, X(u) = x(u), 0 \leq u < s] \\ = P[X(t+s) = l \mid X(s) = k] \end{aligned} \tag{70}$$

for all $s, t \geq 1$, $k, l, x(u) \in [K]$, $0 \leq u < s$. The chain is homogeneous if Eq. 70 is independent of s. The transition probabilities are then denoted:

$$T_{kl}(t) = P[X(t+s) = l \mid X(s) = k]. \tag{71}$$

It can be shown that the transition matrix $T(t)$ is the matrix exponential of a constant rate matrix R times t:

$$T(t) = \exp(Rt) = \sum_{j=0}^{\infty} \frac{1}{j!}(Rt)^j. \tag{72}$$

Example 20 (Jukes–Cantor Model): Consider a fixed position in a DNA sequence, and let $T_{kl}(t)$ be the probability that, due to mutation, nucleotide k changes to nucleotide l after time t at this position (Fig. 8). The Jukes–Cantor model [17] is the simplest DNA substitution model. It assumes that the transition rates from any nucleotide to any other are equal:

$$R = \begin{pmatrix} -3\alpha & \alpha & \alpha & \alpha \\ \alpha & -3\alpha & \alpha & \alpha \\ \alpha & \alpha & -3\alpha & \alpha \\ \alpha & \alpha & \alpha & -3\alpha \end{pmatrix}. \tag{73}$$

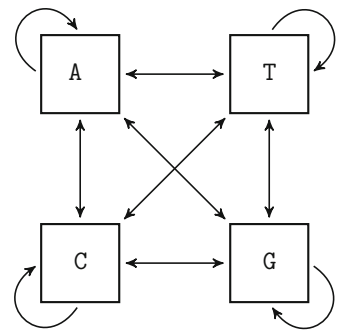


Fig. 8 Nucleotide substitution model. The state space and transitions of a general nucleotide substitution model are shown. For the Jukes–Cantor model (Example 20), all transitions from any nucleotide to any other nucleotide have the same probability $\frac{1}{4}(1 - e^{-4\alpha t})$

The resulting transition matrix $T(t) = \exp(Rt)$ is

$$T(t) = \frac{1}{4}\begin{pmatrix} 1+3e^{-4\alpha t} & 1-e^{-4\alpha t} & 1-e^{-4\alpha t} & 1-e^{-4\alpha t} \\ 1-e^{-4\alpha t} & 1+3e^{-4\alpha t} & 1-e^{-4\alpha t} & 1-e^{-4\alpha t} \\ 1-e^{-4\alpha t} & 1-e^{-4\alpha t} & 1+3e^{-4\alpha t} & 1-e^{-4\alpha t} \\ 1-e^{-4\alpha t} & 1-e^{-4\alpha t} & 1-e^{-4\alpha t} & 1+3e^{-4\alpha t} \end{pmatrix} \tag{74}$$

and the stationary distribution as $t \to \infty$ is uniform, $\pi = (1/4, 1/4, 1/4, 1/4)^t$. □

Example 21 (The Poisson Process): A continuous-time Markov chain $X(t)$ is a counting process, if $X(t)$ represents the total number of events that occur by time t. It is a Poisson process, if in addition $X(0) = 0$, the increments are independent, and in any interval of length t the number of events is Poisson distributed with rate λt:

$$P[X(t+s) - X(s) = k] = P[X(t) = k] = e^{-\lambda t}\frac{(\lambda t)^k}{k!}. \tag{75}$$

The Poisson process is used, for example, to count mutations in a gene. □

Example 22 (Exponential Distribution): The exponential distribution $\mathrm{Exp}(\lambda)$ with parameter $\lambda > 0$ is a common distribution for waiting times. It is defined by the density function:

$$f(x) = \lambda e^{-\lambda x}, \quad \text{for } x \geq 0. \tag{76}$$

If $X \sim \mathrm{Exp}(\lambda)$, then X has expectation $\mathrm{E}(X) = \lambda^{-1}$ and variance $\mathrm{Var}(X) = \lambda^{-2}$. The exponential distribution is memoryless, which means that $P(X > s + t \mid X > t) = P(X > s)$, for all $s, t > 0$. An important consequence of the memoryless property is that the waiting times between successive events are i.i.d. For example, the waiting times τ_n $(n \geq 1)$ between the events of a Poisson process, the sequence of interarrival times, are exponentially distributed, $\tau_n \sim \mathrm{Exp}(\lambda)$, for all $n \geq 1$. □

Exercise 23 (Kimura Model): The Kimura two-parameter model is a DNA substitution model that distinguishes between transitions, i.e., purine-to-purine and pyrimidine-to-pyrimidine substitutions, from transversions, i.e., purine-to-pyrimidine and pyrimidine-to-purine substitutions [18]. It is defined by the rate matrix:

$$R = \begin{pmatrix} -2\beta-\alpha & \beta & \alpha & \beta \\ \beta & -2\beta-\alpha & \beta & \alpha \\ \alpha & \beta & -2\beta-\alpha & \beta \\ \beta & \alpha & \beta & -2\beta-\alpha \end{pmatrix},$$

where $\alpha, \beta \in \mathbb{R}_+$ are the two substitution rates. Assuming that the Markov chain is ergodic, derive its stationary distribution.

6 Hidden Markov Models

A hidden Markov model (HMM) is a statistical model for hidden random variables $Z = (Z_1, \ldots, Z_L)$, which form a homogeneous Markov chain, and observed random variables $X = (X_1, \ldots, X_L)$. Each observed symbol X_n depends on the hidden state Z_n. The HMM is illustrated in Fig. 9. It encodes the following conditional independence statements:

$$Z_{n+1} \perp Z_{n-1} \mid Z_n, \qquad 2 \leq n \leq L-1 \qquad \text{(Markovproperty)} \tag{77}$$

$$X_n \perp X_m \mid Z_n, \qquad 1 \leq m, n \leq L,\ m \neq n \tag{78}$$

The parameters of the HMM consist of the initial state probabilities $\Pi = P(Z_1)$, the transition probabilities $T_{kl} = P(Z_n = l \mid Z_{n-1} = k)$ of the Markov chain, and the emission probabilities $E_{kx} = P(X_n = x \mid Z_n = k)$ of symbols $x \in \mathcal{X}$. The HMM is denoted $\mathrm{HMM}(\Pi, T, E)$. For simplicity, we restrict ourselves here to finite state spaces $\mathcal{Z} = [K]$ of Z and $\mathcal{X}$ of X. The joint probability of (Z, X) factorizes as:

$$\begin{aligned} P(X, Z) &= P(Z_1) \prod_{n=1}^{L-1} P(X_n \mid Z_n) P(Z_{n+1} \mid Z_n) \\ &= \Pi_{Z_1} \prod_{n=1}^{L-1} E_{Z_n, X_n} T_{Z_n, Z_{n+1}}. \end{aligned} \tag{79}$$

The HMM is typically used to model sequence data $x = (x_1, x_2, \ldots, x_L)$ generated by different mechanisms z_n which cannot be observed. Each observation x can be a time series or any other object with a linear dependency structure [19]. In computational biology, the HMM is frequently applied to DNA and protein sequence data, where it accounts for first-order spatial dependencies of nucleotides or amino acids [20].

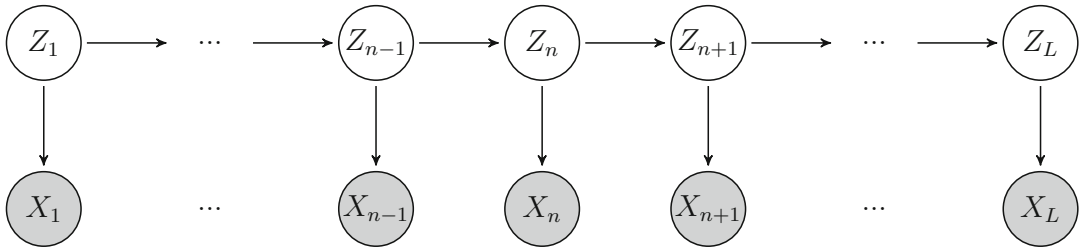


Fig. 9 Hidden Markov model. Shaded nodes represent observed random variables (or symbols) X_n, and clear nodes represent hidden states (or the annotation). Directed edges indicate statistical dependencies which are given, respectively, by transition and emission probabilities among hidden states and between hidden states and observed symbols

Example 24 (CpG Islands): CpG islands are CG-enriched regions in a DNA sequence. They are typically a few hundreds to thousands of base pairs long. We want to use a simple HMM to detect CpG islands in genomic DNA. The hidden states $Z_n \in \mathcal{Z} = \{-, +\}$ indicate whether sequence position n belongs to a CpG island $(+)$ or not $(-)$. The observed sequence is given by the nucleotide at each position, $X_n \in \mathcal{X} = \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$.

Suppose we observe the sequence $x = (\mathtt{C}, \mathtt{A}, \mathtt{C}, \mathtt{G})$. Then, we can calculate the joint probability of x and any state path z by Eq. 79. For example, if $z = (+, -, -, +)$, then $P(X = x, Z = z) = \Pi_+ E_{+,\mathtt{C}} T_{+,-} E_{-,\mathtt{A}} T_{-,-} E_{-,\mathtt{C}} T_{-,+} E_{+,\mathtt{G}}$. □

Typically, one is interested in the hidden state path $z = (z_1, z_2, \ldots, z_L)$ that gave rise to the observation x. For biological sequences, z is often called the annotation of x. In Example 24, the genomic sequence is annotated with CpG islands. For generic parameters, any state path can give rise to a given observed sequence, but with different probabilities. The decoding problem is to find the annotation z^* that maximizes the joint probability:

$$z^* = \operatorname*{argmax}_{z \in \mathcal{Z}} P(X = x, Z = z). \tag{80}$$

There are K^L possible state paths such that already for sequences of moderate length, the optimization problem (Eq. 80) cannot be solved by enumerating all paths.

However, there is an efficient algorithm solving (Eq. 80) based on the following factorization along the Markov chain:

$$\begin{aligned}
\max_Z P(X, Z) &= \max_{Z_1, \ldots, Z_L} P(Z_1) \prod_{n=1}^{L-1} P(X_n \mid Z_n) P(Z_{n+1} \mid Z_n) \\
&= \max_{Z_L} P(Z_L \mid Z_{L-1}) P(X_L \mid Z_L) \\
&\quad \Big[\ldots \Big[\max_{Z_2} P(Z_3 \mid Z_2) P(X_2 \mid Z_2) \\
&\quad \Big[\max_{Z_1} P(Z_2 \mid Z_1) P(X_1 \mid Z_1) \cdot P(Z_1) \Big] \Big] \ldots \Big].
\end{aligned} \tag{81}$$

Thus, the maximum over state paths $(Z_1, \ldots, Z_L)$ can be obtained by recursively computing maxima over each Z_n. Each of the L terms in parenthesis defines a probability distribution over K states by maximizing over K values. Hence, the time complexity of the algorithm is $O(LK^2)$, despite the fact that the maximum is over K^L paths. This procedure is known as dynamic programming and it is the workhorse of biological sequence analysis. For HMMs, it is known as the Viterbi algorithm [21].

In order to compute the marginal likelihood $P(X = x)$ of an observed sequence x, we need to sum the joint probability $P(Z = z, X = x)$ over all hidden states $z \in \mathcal{Z}$. The length of this sum is K^L, but it can be computed efficiently by the same dynamic programming principle used for the Viterbi algorithm:

$$\begin{aligned}\sum_{Z} P(X,Z) &= \sum_{Z_1,\ldots,Z_L} P(Z_1)\prod_{n=1}^{L-1} P(X_n \mid Z_n)P(Z_{n+1} \mid Z_n) \\ &= \sum_{Z_L} P(Z_L \mid Z_{L-1})P(X_L \mid Z_L) \\ &\quad \Big[\ldots\Big[\sum_{Z_2} P(Z_3 \mid Z_2)P(X_2 \mid Z_2) \\ &\quad \Big[\sum_{Z_1} P(Z_2 \mid Z_1)P(X_1 \mid Z_1)\cdot P(Z_1)\Big]\Big]\ldots\Big].\end{aligned} \tag{82}$$

Indeed, this factorization is the same as in Eq. 81 with maxima replaced by sums. The recursive algorithm implementing (Eq. 82) is known as the forward algorithm. In each step, it computes the partial solution $f(n, Z_n) = P(X_1, \ldots, X_n, Z_n)$.

The factorization along the Markov chain can also be done in the other direction starting the recursion from Z_L down to Z_1. The resulting backward algorithm generates the partial solutions $b(n, Z_n) = P(X_{n+1}, \ldots, X_L \mid Z_n)$. From the forward and backward quantities, one can also compute the position-wise posterior state probabilities:

$$\begin{aligned}P(Z_n \mid X) &= \frac{P(X,Z_n)}{P(X)} = \frac{P(X_1,\ldots,X_n,Z_n)P(X_{n+1},\ldots,X_L \mid Z_n)}{P(X)} \\ &= \frac{f(n,Z_n)b(n,Z_n)}{P(X)}.\end{aligned} \tag{83}$$

For example, in the CpG island HMM (Example 24), we can compute, for each nucleotide, the probability that it belongs to a CpG island given the entire observed DNA sequence. Selecting the state that maximizes this probability independently at each sequence position is known as posterior decoding. In general, the result will be different from Viterbi decoding.

Example 25 (Pairwise Sequence Alignment): The pair HMM is a statistical model for pairwise alignment of two observed sequences over a fixed alphabet $\mathcal{A}$. For protein sequences, $\mathcal{A}$ is the set of 20 natural amino acids and for DNA sequences, $\mathcal{A}$ consists of the four nucleotides, plus the gap symbol ("-"). At each position of the alignment, a hidden variable $Z_n \in \mathcal{Z} = \{\mathtt{M}, \mathtt{X}, \mathtt{Y}\}$ indicates whether

there is a (mis-)match (M), an insertion (X), or a deletion (Y) in sequence y relative to sequence x. For example:

$z =$ MMMMMMMMMMMMMXXMMMMMMMMMMMMMYMMMMYMMMMM
$x =$ CTRPNNNTRKSIRPQIGPGQAFYATGD–IGDI–RQAHC
$y =$ CGRPNNHRIKGLR––IGPGRAFFAMGAIRGGEIRQAHC

The emitted symbols are pairs (X_n, Υ_n) of aligned sequence characters with state space $(\mathcal{A} \times \mathcal{A}) \backslash \{(-, -)\}$. Thus, a pairwise alignment is a probabilistically generated sequence of pairs of symbols.

The choice of transition and emission probabilities corresponds to fixing a scoring scheme in nonprobabilistic formulations of sequence alignment. For example, the emission probabilities $P[(a, b) \mid \mathtt{M}]$ from a match state encode pairwise amino acid preferences and can be modeled by substitution matrices, such as PAM and BLOSUM [20].

In the pair HMM, computing an optimal alignment between x and y means to find the most probable state path $z^* = \operatorname{argmax}_z P(X = x, \Upsilon = y, Z = z)$, which can be solved using the Viterbi algorithm. Using the forward algorithm, we can also compute efficiently the marginal probability of two sequences being related independent of their alignment, $P(X, \Upsilon) = \Sigma_Z P(X, \Upsilon, Z)$. In general, this probability is more informative than the posterior $P(Z \mid X, \Upsilon)$ of an optimal alignment z^* because many alignments tend to have the same or nearly the same probability such that $P(Z = z^* \mid X, \Upsilon)$ can be very small. Finally, we can also compute the probability of two characters x_n and y_m being aligned by means of posterior decoding. □

Example 26 (Profile HMM): Profile hidden Markov models represent groups of related sequences, such as protein families. They are used for searching homologous sequences and for building multiple sequence alignments. They can be regarded as unrolled versions of the pair HMM. A profile HMM is a statistical model for observed sequences, which are regarded as i.i.d. realizations. It has site-specific emission probabilities $E_n(a) = P(X_n = a)$. In its simplest form allowing only gap-free alignments, the probability of an observation x is just

$$P(X = x) = \prod_{n=1}^{L} E_n(x_i). \tag{84}$$

The matrix $(E_n(a))_{1 \le n \le L,\ a \in \mathcal{A}}$ is called a position-specific scoring matrix (PSSM).

Profile HMMs can also model indels. Figure 10 shows the hidden state space of such a model. It has match states M_n, which can emit symbols according to the probability tables E_n, insert states I_n, which usually emit symbols in an unspecific manner, and delete states D_n, which do not emit any symbols. The possible transitions between those states allow for modeling alignment gaps of any length.

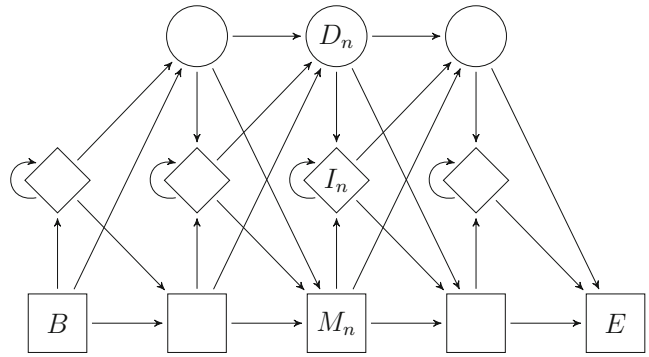


Fig. 10 Profile hidden Markov model. The hidden state space and its transitions are shown for the profile HMM of length $L = 3$. Match states are denoted M_n, insert states I_n, and delete states D_n. B and E denote silent begin and end states, respectively. With match and insert states, probability tables for the emissions of symbols (amino acids or nucleotides, and gaps) are associated

A given profile HMM for a protein family can be used to detect new sequences that belong to the same family. For a query sequence x, we can either consider the most probable alignment of the sequence to the HMM, $P(X = x, Z = z^*)$, or the marginal probability independent of the alignment, $P(X = x) = \Sigma_Z P(X = x, Z)$, to decide about family membership. □

Parameter estimation in HMMs is complicated by the presence of hidden variables. In Subheading 2, the EM algorithm has been introduced for finding a local maximum of the likelihood surface. For HMMs, the EM algorithm is known as the Baum–Welch algorithm [22]. For simplicity, let us ignore the initial state probabilities Π and summarize the parameters of the HMM by $\theta = (T, E)$. For ML estimation, we need to maximize the observed log-likelihood:

$$\begin{aligned} \ell_{\text{obs}}(\theta) &= \log P(X \mid \theta) = \log \sum_{Z} P(X, Z \mid \theta) \\ &= \log \sum_{Z^{(1)}, \ldots, Z^{(N)}} \prod_{i=1}^{N} P(X^{(i)}, Z^{(i)} \mid \theta), \end{aligned} \tag{85}$$

where $X^{(1)}, \ldots, X^{(N)}$ are the i.i.d. observations. For each observation, we can rewrite the joint probability as:

$$P(X^{(i)}, Z^{(i)} \mid \theta) = \prod_{k \in [K]} \prod_{x \in \mathcal{X}} E_{kx}^{N_{kx}(Z^{(i)})} \cdot \prod_{k \in [K]} \prod_{l \in [K]} T_{kl}^{N_{kl}(Z^{(i)})}, \tag{86}$$

where $N_{kx}(Z^{(i)})$ is the number of x emissions when in state k and $N_{kl}(Z^{(i)})$ the number of k-to-l transitions in state path $Z^{(i)}$ (cf. Eq. 68).

In the E step, the expectation of Eq. 85 is computed with respect to $P(Z \mid X, \theta')$, where θ' is the current best estimate of θ. We use Eq. 86 and denote by N_{kx} and N_{kl} the expected value of $\Sigma_i N_{kx}(Z^{(i)})$ and $\Sigma_i N_{kl}(Z^{(i)})$, respectively, to obtain

$$\begin{aligned}
\mathrm{E}[\ell_{hid}(\theta)] &= \sum_{Z} P(Z \mid X, \theta') \log P(X, Z \mid \theta) \\
&= \sum_{Z^{(1)}, \ldots, Z^{(N)}} P(Z \mid X, \theta') \\
&\quad \left[\sum_{k,\ x} N_{kx}(Z^{(i)}) \log E_{kx} + \sum_{k,\ l} N_{kl}(Z^{(i)}) \log T_{kl} \right] \\
&= \sum_{k,\ x} N_{kx} \log E_{kx} + \sum_{k,\ l} N_{kl} \log T_{kl}.
\end{aligned} \tag{87}$$

The expected counts N_{kx} and N_{kl} are the sufficient statistics [11] of the HMM, i.e., with respect to the model, they contain all information about the parameters available from the data. The expected counts can be computed using the forward and backward algorithms. In the M step, this expression is maximized with respect to $\theta = (T,\ E)$. We find the MLEs $\hat{T}_{kl} = N_{kl} / \sum_m N_{km}$ and $\hat{E}_{kx} = N_{kx} / \sum_y N_{ky}$.

7 Bayesian Networks

Bayesian networks are a class of probabilistic graphical models which generalize Markov chains and HMMs. The basic idea is to use a graph for encoding conditional independences among random variables (Fig. 11). The graph representation provides not only an intuitive and simple visualization of the model structure, but it is also the basis for designing efficient algorithms for inference and learning in graphical models [23–25].

A Bayesian network (BN) for a set of random variables $X = (X_1, \ldots, X_L)$ consists of a directed acyclic graph (DAG) and local probability distributions (LPDs). The DAG $G = (V, E)$ has vertex set $V = [L]$ and edge set $E \subseteq V \times V$. Each vertex $n \in V$ is identified with the random variable X_n. If there is an edge $X_m \to X_n$ in G, then X_m is a parent of X_n and X_n is a child of X_m. For each vertex $n \in V$, there is an LPD $P(X_n \mid X_{\mathrm{pa}(n)})$, where $\mathrm{pa}(n)$ is the set of parents of X_n in G. The Bayesian network model is defined as the family of distributions for which the joint probability of X factors into conditional probabilities as:

$$P(X_1, \ldots, X_L) = \prod_{n=1}^{L} P(X_n \mid X_{\mathrm{pa}(n)}). \tag{88}$$

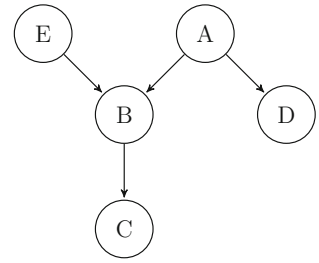


Fig. 11 Example of a Bayesian network. Vertices correspond to random variables and edges represent conditional probabilities. The graph encodes conditional independence statements about the random variables U, V, W, X, Y, and Z. Their joint probability factors according to the graph as $P(U, V, W, X, Y) = P(U)P(Y)P(V \mid U, Y)P(W \mid V)P(X \mid U)$

In this case, we write $X \sim \mathrm{BN}(G, \theta)$, where $\theta = (\theta_1, \ldots, \theta_L)$ denotes the parameters of the LPDs.

For the Bayesian network shown in Fig. 11, we find $P(U, V, W, X, \Upsilon) = P(U)P(\Upsilon)P(V \mid U, \Upsilon)P(W \mid V)P(X \mid U)$. The graph encodes several conditional independence statements about (U, V, W, X, Υ), including, for example, $W \perp \{U, X\} \mid V$.

Example 27 (Markov Chain): A finite Markov chain is a Bayesian network with the DAG $X_1 \to X_2 \to \cdots \to X_L$, denoted C, and joint distribution:

$$P(X_1, \ldots, X_n) = P(X_1)P(X_2 \mid X_1)P(X_3 \mid X_2) \cdots P(X_L \mid X_{L-1}). \tag{89}$$

If $X \sim \mathrm{MC}(\Pi, T)$ is homogeneous, then the LPDs are $\theta_1 = P(X_1) = \Pi$ and $\theta_{n+1} = P(X_{n+1} \mid X_n) = T$ for all $n \in [L-1]$ such that $\mathrm{MC}(\Pi, T) = \mathrm{BN}(C, \theta)$. Similarly, HMMs are Bayesian networks with hidden variables Z and factorized joint distribution given in Eq. 79. □

The meaning of the parameters θ of a Bayesian network depends on the family of distributions that has been chosen for the LPDs. In the general case of a discrete random variable with finite state space, θ_n is a conditional probability table. If each vertex X_n has K possible states, then:

$$\theta_n = \left(P(X_n = a \mid X_{\mathrm{pa}(n)} = b)\right)_{b \in [K]^{pa(n)},\, a \in [K]} \tag{90}$$

has $K^{\mathrm{pa}(n)} \times (K-1)$ free parameters. If X_n depends on all other variables, then θ_n has the maximal number of $K^L - 1$ parameters, which is exponential in the number of vertices. If, on the other hand, X_n is independent of all other variables, $\mathrm{pa}(n) = \emptyset$, then θ_n has $(K-1)$ parameters, which is independent of L. For the chain (Example 27) where each vertex has exactly one outgoing and one incoming edge, we find a total of $(K-1) + (L-1)K(K-1)$ free parameters which is of order $O(LK^2)$.

A popular model for continuous random variables X_n is the linear Gaussian model. Here, the LPDs are Gaussian distributions with mean a linear function of the parents:

$$P(X_n \mid X_{\mathrm{pa}(n)}) = \mathrm{Norm}(\nu_n + w_n^t \cdot X_{\mathrm{pa}(n)},\ \sigma_n^2), \tag{91}$$

with parameters $\nu_n \in \mathbb{R}$ and $w_i \in \mathbb{R}^{\mathrm{pa}(n)}$ specifying the mean and variance σ_n^2. The number of parameters increases linearly with the number of parents, but only linear relationships can be modeled. All marginal and conditional probabilities of $(X_1, \ldots, X_L)$ are also Gaussians.

Learning a Bayesian network $\mathrm{BN}(G, \theta)$ from data $\mathcal{D}$ can be done in different ways following either the Bayesian or the maximum likelihood approach as introduced in Subheading 2. In general, it involves first finding the optimal network structure:

$$G^* = \underset{G}{\mathrm{argmax}}\ P(G \mid \mathcal{D}), \tag{92}$$

and then estimating the parameters:

$$\theta^* = \underset{\theta}{\mathrm{argmax}}\ P(\theta \mid G^*,\ \mathcal{D}) \tag{93}$$

for the given optimal structure G^*. The first step is a model selection problem as introduced in Subheading 2.

Model selection for Bayesian networks is a particularly hard problem because the number of DAGs increases super-exponentially with the number of vertices rendering exhaustive searches impractical, and the objective function in Eq. 92 is difficult to compute. Recall that the posterior $P(G \mid \mathcal{D})$ is proportional to the product $P(\mathcal{D} \mid G)P(G)$ of marginal likelihood and network prior, and the marginal likelihood:

$$P(\mathcal{D} \mid G) = \int P(\mathcal{D} \mid \theta,\ G) P(\theta \mid G)\ d\theta \tag{94}$$

is usually analytically intractable. Here, $P(\theta \mid G)$ is the prior distribution of parameters given the network topology.

To address this limitation, the marginal likelihood (Eq. 94) can be approximated by a function that is easier to evaluate. A popular choice is the Bayesian information criterion (BIC) [26]:

$$\log P(\mathcal{D} \mid G) \approx \log P(\mathcal{D} \mid \hat{\theta}_{\mathrm{ML}}, G) - \frac{1}{2}\nu \log N, \tag{95}$$

where ν is the number of free parameters of the model and N the size of the data. The BIC approximation can be derived under certain assumptions, including a unimodal likelihood. It replaces computation of the integral (Eq. 94) by evaluating the integrand at the MLE and adding the correction term $-(\nu \log N)/2$, which penalizes models of high complexity.

The model selection problem remains hard even with a tractable scoring function, such as BIC, because of the enormous search

space. Local search methods, such as greedy hill climbing or simulated annealing, are often used in practice. They return a local maximum as a point estimate for the best network structure. Results can be improved by running several local searches from different starting topologies.

Often, data are sparse and we will find diffuse posterior distributions of network structures, which might not be represented very well by a single point estimate. In the fully Bayesian approach, we aim at estimating the full posterior $P(G \mid \mathcal{D}) \propto P(\mathcal{D} \mid G)P(G)$. One way to approximate this distribution is to draw a finite number of samples from it. Markov chain Monte Carlo (MCMC) methods generate such a sample by constructing a Markov chain that converges to the target distribution [27].

In the Metropolis–Hastings algorithm [28], we start with a random DAG $G^{(0)}$ and then iteratively generate a new DAG $G^{(n)}$ from the previous one $G^{(n-1)}$ by drawing it from a proposal distribution Q:

$$G^{(n)} \sim Q(G^{(n)} \mid G^{(n-1)}). \tag{96}$$

The new DAG is accepted with acceptance probability:

$$\min\left\{\frac{P(\mathcal{D} \mid G^{(n)})P(G^{(n)})Q(G^{(n-1)} \mid G^{(n)})}{P(\mathcal{D} \mid G^{(n-1)})P(G^{(n-1)})Q(G^{(n)} \mid G^{(n-1)})}, 1\right\} \tag{97}$$

Otherwise, the model is left unchanged and the next sample is drawn. With this acceptance probability, it is guaranteed that the Markov chain is ergodic and converges to the desired distribution. After an initial burn-in phase, samples from the stationary phase of the chain are collected, say $G^{(m)}, \ldots, G^{(N)}$. Any feature f of the network (e.g., the presence of an edge or a subgraph) can be estimated as the expected value:

$$\mathrm{E}(f) = \sum_{G} f(G)P(G \mid \mathcal{D}) \approx \frac{1}{N}\sum_{n=m}^{N} f(G^{(n)}). \tag{98}$$

A critical point of the Metropolis–Hastings algorithm is the choice of the proposal distribution Q, which encodes the way the network space is explored. Because not all graphs, but only DAGs, are allowed, computing the transition probabilities $Q(G^{(n)} \mid G^{(n-1)})$ is usually the main computational bottleneck.

Parameter estimation, i.e., solving (Eq. 93), can be done along the lines described in Subheading 2 following either the ML or the Bayesian approach. If the model contains hidden random variables, then the EM algorithm (Subheading 3) can be used. However, this approach is feasible only if efficient inference algorithms are available. For hidden Markov models (Subheading 6), the forward and backward algorithms provided an efficient way to compute marginal probabilities and the expected hidden log-likelihood. These

algorithms can be generalized to the sum–product algorithm for tree-like graphs and the junction tree algorithm for general DAGs. The computational complexity of the junction tree algorithm is exponential in the size of the largest clique of the so-called moralized graph, which is obtained by dropping edge directions and adding edges between any two vertices that have a common child in the original DAG [11].

Alternatively, if exact inference is computationally too expensive, then approximate inference can be used. For example, Gibbs sampling [29] is an MCMC technique for generating a sample from the joint distribution $P(X_1, \ldots, X_L)$. The idea is to iteratively sample from the conditional probabilities of $P(X_1, \ldots, X_L)$, starting with $X_1^{(n+1)} \sim P(X_1 \mid X_2^{(n)}, \ldots, X_L^{(n)})$ and cycling through all variables in turns:

$$X_j^{(n+1)} \sim P(X_j \mid X_1^{(n+1)}, \ldots, X_{j-1}^{(n+1)}, X_{j+1}^{(n)}, \ldots, X_L^{(n)}) \quad \text{for all } j = 2, \ldots, L. \tag{99}$$

Gibbs sampling can be regarded as a special case of the Metropolis–Hastings algorithm. It is particularly useful, if it is much easier to sample from the conditionals $P(X_k \mid X_{\setminus k})$ than from the joint distribution $P(X_1, \ldots, X_L)$, where $X_{\setminus k}$ denotes all variables X_n except X_k. For graphical models, the conditional probability of each vertex X_k depends only on its Markov blanket $X_{\mathrm{MB}}(k)$, defined as the set of its parents, children, and co-parents (vertices with the same children), $P(X_k \mid X_{\setminus k}) = P(X_k \mid X_{\mathrm{MB}(k)})$.

Example 28 (Phylogenetic Tree Models): A phylogenetic tree model [30] for a set of aligned DNA sequences from different species is a Bayesian network model, where the graph is a tree in which the leaves represent the observed contemporary species and the interior vertices correspond to common extinct ancestors (Fig. 12). The topology (graph structure) S defines the branching order and the branch lengths correspond to (phylogenetic) time. The LPDs are defined by a nucleotide substitution model (Subheading 5).

Let $X^{(i)} \in \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}^L$ denote the i-th column of a multiple sequence alignment of L observed species. We regard the alignment columns as independent observations of the evolutionary process. The character states of the hidden (extinct) ancestors are denoted $Z^{(i)}$. The likelihood of the observed sequence data $X = (X^{(1)}, \ldots, X^{(N)})$ given the tree topology S and the branch lengths t is

$$P(X \mid S, t) = \sum_Z \prod_{i=1}^N P(X^{(i)}, Z^{(i)} \mid S, t), \tag{100}$$

where $P(X^{(i)}, Z^{(i)} \mid S, t)$ factors into conditional probabilities according to the tree structure. This marginal probability can be computed efficiently with an instance of the sum–product algorithm known as the peeling algorithm (or Felsenstein algorithm) [31].

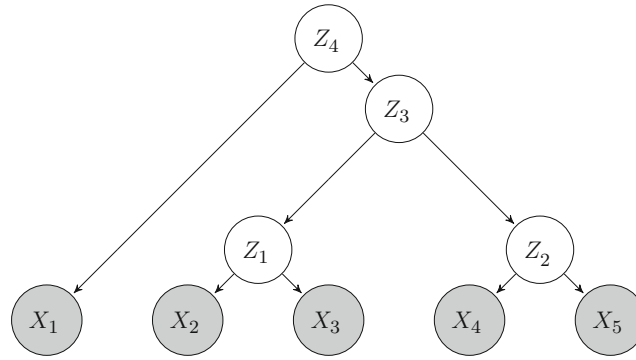


Fig. 12 Phylogenetic tree model. The observed random variables X_i represent contemporary species and the hidden random variables Z_i their unknown common ancestors

For example, in the tree displayed in Fig. 12, each observation X has probability:

$$P(X) = \sum_{Z} P(X, Z) \tag{101}$$

$$\begin{aligned} &= \sum_{Z} P(X_1 \mid Z_4) P(X_2 \mid Z_1) P(X_3 \mid Z_1) P(X_4 \mid Z_2) \cdot \\ &\quad \cdot P(X_5 \mid Z_2) P(Z_1 \mid Z_3) P(Z_2 \mid Z_3) P(Z_3 \mid Z_4) P(Z_4) \end{aligned} \tag{102}$$

$$\begin{aligned} &= \sum_{Z_4} P(Z_4) P(X_1 \mid Z_4) \left[\sum_{Z_3} P(Z_3 \mid Z_4) \right. \\ &\quad \left[\sum_{Z_2} P(Z_2 \mid Z_3) P(X_4 \mid Z_2) P(X_5 \mid Z_2) \right] \\ &\quad \left. \cdot \left[\sum_{Z_1} P(Z_1 \mid Z_3) P(X_2 \mid Z_1) P(X_3 \mid Z_1) \right] \right], \end{aligned} \tag{103}$$

where we have omitted the dependency on the branch length t. Several software packages implement ML or Bayesian learning of phylogenetic tree models. □

In the simplest case, we suppose that the observed alignment columns are independent. However, it is more realistic to assume that nucleotide substitution rates vary across sites because of varying selective pressures. For example, there could be differences between coding and noncoding regions, among different regions of a protein (loops, and catalytic sites), or among the three bases of a triplet coding for an amino acid. More sophisticated models can account for this rate heterogeneity. Let us assume site-specific substitution rates r_i such that the local probabilities become

$P(X^{(i)} \mid r_i, t, S)$. To model the distribution of the rates, often a gamma distribution is used.

Example 29 (Gamma Distribution): The gamma distribution $\text{Gamma}(\alpha, \beta)$ is parametrized by a shape parameter α and a rate parameter β. It is defined by the density function:

$$f(x) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x}, \quad \text{for } x \geq 0. \tag{104}$$

Its expectation is $\text{E}(X) = \alpha/\beta$ and its variance $\text{Var}(X) = \alpha/\beta^2$. The gamma distribution generalizes several other distributions, for example $\text{Gamma}(1, \lambda) = \text{Exp}(\lambda)$ (Example 22). □

Another approach to account for varying mutation rates are phylogenetic hidden Markov models (phylo-HMMs).

Example 30 (Phylo-HMM): Phylo-HMMs [32] combine HMMs and phylogenetic trees into a single Bayesian network model. The idea is to use an HMM along the linear chain of the genomic sequence and, at each position, to condition a phylogenetic tree model on the hidden state (Fig. 13). This architecture allows for modeling different evolutionary histories at different sites of the genome. In particular, the model can account for heterogeneity in the rate of evolution, for example, due to functionally conserved elements, but it also allows for a change in tree topology along the sequence, a situation that can result from recombination [23]. Phylo-HMMs are also used for gene finding. □

Exercise 31 (Inference in Bayesian Networks): Consider the gene network on five genes denoted A, B, C, D, E, with the graph structure displayed below. Gene expression profiles under different conditions C1–C9 have been observed and are summarized in the table below, where a zero indicates that the gene is not expressed and a one that it is expressed.

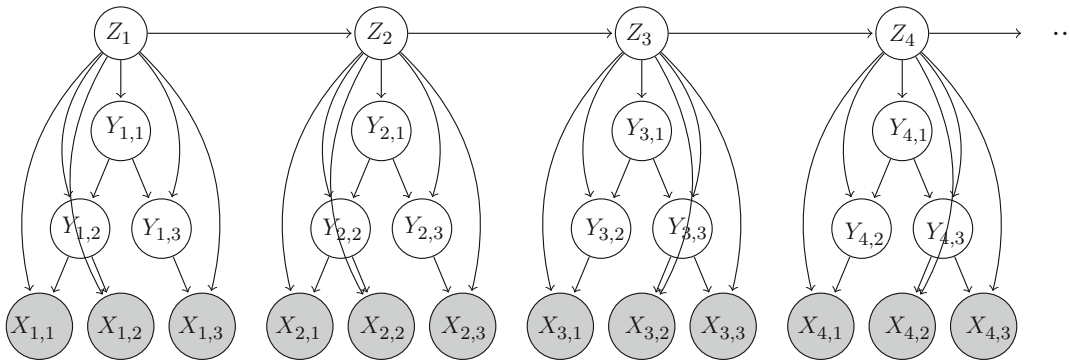
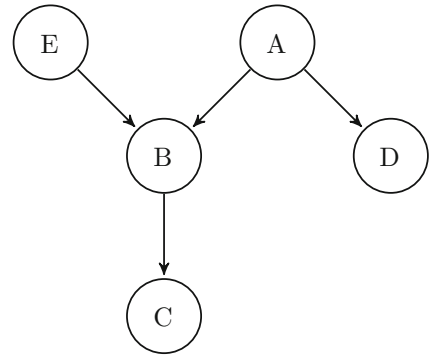


Fig. 13 Phylo-HMM. Shown are the first four positions of a Phylo-HMM. The hidden Markov chain has random variables Z. In the trees, Y denote the hidden common ancestors and X the observed species. Note that the tree topology changes between position 2 and 3

	A	B	C	D	E
C1	0	0	0	0	0
C2	0	0	0	0	1
C3	0	0	0	0	1
C4	1	1	1	1	0
C5	1	0	1	1	0
C6	0	0	0	1	1
C7	1	1	1	1	0
C8	1	0	0	0	1
C9	1	0	0	1	1



(a) Specify the adjacency matrix of the directed graph.

(b) Determine the local probability distributions for each vertex of the graph. Use conditional counting to determine the conditional probabilities as:

$$P(X_i \mid X_{\mathrm{pa}(i)}) \approx \frac{N(X_i, X_{\mathrm{pa}(i)})}{\sum_k N(X_i = k,\ X_{\mathrm{pa}(i)})},$$

where $N(X_i,\ X_{\mathrm{pa}(i)})$ is the number of joint observations of X_i and its parents.

(c) What is the joined probability of $(X_{\mathrm{A}}, X_{\mathrm{B}}, X_{\mathrm{C}}, X_{\mathrm{D}}, X_{\mathrm{E}})$ for this network?

(d) We now want to determine the most probable explanation for observing a gene C to be active as a result of the influences of its upstream genes A and E. For this, one has to infer the posterior probabilities $P(\mathrm{A} \mid \mathrm{C} = 1)$ and $P(\mathrm{E} \mid \mathrm{C} = 1)$ using Bayes theorem. Here, assume that the probabilities $P(\mathrm{A})$ and $P(\mathrm{E})$ derived from the expression data are suitable prior probabilities. Which constellation is most likely to trigger the expression of C?

References

1. Ewens WJ, Grant GR (2005) Statistical methods in bioinformatics: an introduction, 2nd edn. Springer, Berlin
2. Deonier RC, Tavaré S, Waterman MS (2005) Computational genome analysis: an introduction. Springer, Berlin
3. Davison AC (2009) Statistical models. Cambridge series in statistical and probabilistic mathematics. Cambridge University Press, Cambridge
4. Ross SM (2007) Introduction to probability models. Academic, London

5. Hardy GH (1908) Mendelian proportions in a mixed population. Science 28:49–50
6. Weinberg W (1908) Über den Nachweis der Vererbung beim Menschen. Jahres Wiertt Ver Vaterl Natkd 64:369–382
7. Pachter L, Sturmfels B (2005) Algebraic statistics for computational biology. Cambridge University Press, Cambridge
8. Casella G, Berger RL (2002) Statistical Inference. Thomson Learning, Pacific Grove
9. Efron B, Tibshirani R (1993) An introduction to the bootstrap. Chapman and Hall/CRC, Boca Raton
10. Gelman A, Carlin JB, Stern HS, Rubin DB (2004) Bayesian data analysis, vol 46, 2 edn. Chapman and Hall/CRC, Boca Raton
11. Bishop CM (2006) Pattern recognition and machine leaning. Springer, Berlin
12. Kass R, Raftery A (1995) Bayes factors. J Am Stat Assoc 90:773–795
13. Dempster AP, Laird NM, Rubin DB (1977) Maximum likelihood from incomplete data via the EM algorithm. J R Stat Soc Ser B 39:1–38
14. Norris JR (1998) Markov chains. Cambridge University Press, Cambridge
15. Wright S (1990) Evolution in Mendelian populations. Bull Math Biol 52:241–295
16. Fisher RA (1930) The genetical theory of natural selection. Clarendon Press, Oxford
17. Jukes TH, Cantor CR (1969) Evolution of protein molecules. Mamm Protein Metab 3:21–132
18. Kimura M (1980) A simple method for estimating evolutionary rates of base substitutions through comparative studies of nucleotide sequences. J Mol Evol 16:111–120
19. Rabiner LR (1989) A tutorial on HMM and selected applications in speech recognition. Proc IEEE 77:257–286
20. Durbin R (1998) Biological sequence analysis. Probabilistic models of proteins and nucleic acids. Cambridge University Press, Cambridge
21. Viterbi A (1967) Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. IEEE Trans Inf Theory 13:260–269
22. Baum LE (1972) An inequality and associated maximization technique in statistical estimation for probabilistic functions of Markov processes. Inequalities 3:1–8
23. Husmeier D, Dybowski R, Roberts S (2005) Probabilistic modeling in bioinformatics and medical informatics. Springer, New York
24. Koller D, Friedman N (2009) Probabilistic graphical models: principles and techniques. The MIT Press, Cambridge
25. Jordan MI (1998) Learning in graphical models. Kluwer Academic Publishers, Dordrecht
26. Schwarz G (1978) Estimating the dimension of a model. Ann Stat 6:461–464
27. Neal RM (1993) Probabilistic inference using Markov Chain Monte Carlo methods. Intelligence 62:144
28. Hastings WK (1970) Monte Carlo sampling methods using Markov chains and their applications. Biometrika 57:97–109
29. Geman S, Geman D (1984) Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. IEEE Trans Pattern Anal Mach Intell 6:721–741
30. Felsenstein J (2004) Inferring phylogenies. Sinauer Associates, Sunderland
31. Felsenstein J (1981) Evolutionary trees from DNA sequences: a maximum likelihood approach. J Mol Evol 17:368–376
32. Siepel A, Haussler D (2005) Phylogenetic hidden Markov models. Statistical methods in molecular evolution. Springer, New York, pp 325–351