Bayesian Inference for Species Distribution Modelling with Gaussian Processes

Isaac William Caruso

Amherst College
Department of Computer Science

2021-04-08

Contents

Chapter 1

Probability and Bayes' theorem

This chapter provides an introduction to the probability concepts necessary to understand Bayesian inference. Simply put, Bayesian inference is a statistical technique for estimating a quantity of interest upon the observation of data, while explicitly incorporating prior knowledge or belief about that quantity of interest. Before embarking on an exposition of Bayesian statistics, we must first gain a basic understanding of a few key elements of probability theory—random variables, cumulative distribution functions, probability functions/ distributions, expected value, and conditional probability. This chapter then concludes by introducing Bayes' theorem and Bayesian inference, and demonstrating the steps for performing a simple Bayesian update.

1.1 Random Variables

Imagine for a moment you are tossing a fair coin. There are many experiments you could perform by tossing a coin, but let us consider our quantity of interest to be the fraction of times our coin lands on a tails. It is clear that the number of tails, the outcome of our experiment, is dependent on the eventual realization of some random process. A random variable

X is a variable whose value is dependent on the outcome(s) of a stochastic phenomenon. The realized value of X is denoted as x.

In the example of tossing a coin, where the data is a sequence of coin tosses, e.g., $[H, T, T, ...]$, we define the random variable X to be the number of tails. If we toss the coin twice, X has three possible realized states, x, depending on the outcome of this stochastic experiment: $x = 0$, $x = 1$, or $x = 2$. Table 1.1 shows the probability that our random variable X takes value x, some actual number of tails.

Table 1.1: P(X = x) for two tosses

x	P(X = x)
0	0.25
1	0.50
2	0.25

The r.v. X is is an example of a *discrete random variable*. Discrete random variables can only assume discrete values. To the contrary, continuous random variables are useful for describing continuous sample spaces. For example, a continuous random variable may be used to represent the outcome of an experiment measuring blossoming heights of flowers, where the data is a sequence of observations of heights at which different flowers blossomed. In this case, the outcome of our blossoming experiment can be any of an infinite number of real values, thus is properly modeled by a continuous random variable.

1.2 Cumulative Distribution Functions

In the previous example we represented the probability of various outcomes of a coin toss experiment in a tabular format. Another way to represent this set of probabilities is as a *cumulative distribution function*.

Definition 1.1 (Cumulative distribution function 'CDF'). *The cumulative distribution function is defined as a function where* $F_X \in [0, 1]$*:*

$$F_X(x) \doteq P(X \leq x)$$

The cumulative distribution function $F_X(x)$ simply represents the probability that a random variable X takes a value less than or equal to x for each possible input value of x. Figure 1.1 depicts a graphical representation of the CDF for our coin tossing experiment.

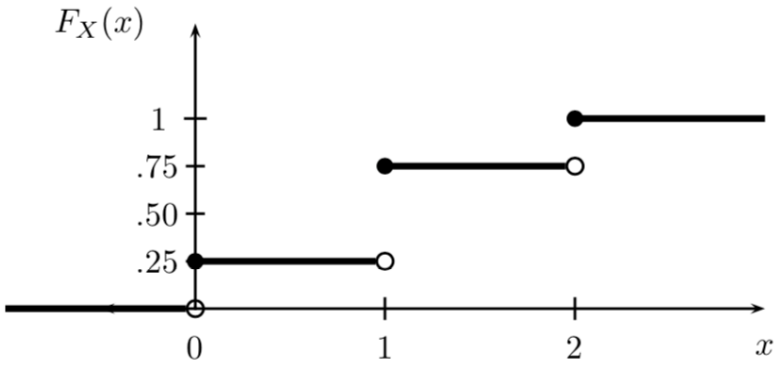

Figure 1.1: CDF for tossing a coin twice (Wasserman, 2004)

Here, for every value of x—the number of tails in two coin tosses—the probability that X is equal to or less than this value is represented. In this discrete example, we see that our CDF is represented by several non-decreasing discrete lines defined for all x. In the example of a continuous random variable, this function is a left-continuous, non-decreasing distribution also defined for all x.

1.3 Probability Mass and Density Functions

The probability mass function and probability density function allow us to express probabilities of events over a sample space, and find their use with discrete and continuous random

variables respectively.

In the discrete setting, the probability mass function for a random variable X yields the probability that X takes a value for every possible value that X can take.

Definition 1.2 (Probability mass function 'PMF')**.** *The probability function for a discrete random variable X—the probability mass function for X—is defined as a function*

$$f_X(x) \doteq P(X = x)$$

Here, the PMF has a few key attributes. Namely $P(X = x) > 0$ for every x in the sample space S_X of X, and $\sum_{x \in S_X} f_X(x) = 1$. With these features in mind, the probability mass function of X follows logically from the cumulative distribution function of X insofar as the CDF is the sum of the PMF for all $x_i \leq x$, i.e.,

$$F_X(x) \doteq P(X \leq x) = \sum_{x_i \leq x} f_X(x_i).$$

In the case where the random variable X is continuous, its PDF is defined as follows.

Definition 1.3 (Probability density function 'PDF')**.** *The probability function for a continuous random variable X—the probability density function for X—is defined as a function $f(x)$ where a and b are two real numbers such that $a \leq b$, so*

$$P(a < X < b) \doteq \int_a^b f_X(x)dx.$$

In other words, the probability that the realized value x of our continuous random variable X is between two numbers a and b is equal to the integral of the probability density function of x from $x = a$ to $x = b$. This formalization of the PDF $f_X(x)$ allows a natural comparison with the CDF $F_X(x)$ of X,

$$F_X(x) \doteq \int_{-\infty}^{x} f_X(x)dx.$$

Specifically, this implies that $F'_X(x) = f_X(x)$ for all differentiable points of F_X. In plain English this signifies that the derivative of the CDF is the PDF.

1.4 Expectation

One of the final core statistical concepts necessary to approach Bayesian statistics on sure footing is the idea of expectation or expected value.

Definition 1.4 (Expectation)**.** *The expectation or expected value of a random variable X is*

$$E(X) \doteq \begin{cases} \sum_x x f_X(x) & \text{if } X \text{ is discrete} \\ \int x f_X(x)dx & \text{if } X \text{ is continuous} \end{cases}$$

To return to a familiar example, consider a random variable X to represent the number of tails in 6 coin tosses. Figure 1.2 depicts the PMF for X using the *binomial distribution* $B(6, 0.5)$ which represents the probability of observing a specific number of successes in a success-failure experiment.

In this case, the x-axis represents each possible outcome x and the y-axis is the probability of that outcome. Table 1.2 presents the value of the binomial PMF for every $x \in X$.

Computing $E(X)$ given the values in this table is demonstrated using the discrete case of Definition 1.4 in Example 1.1.

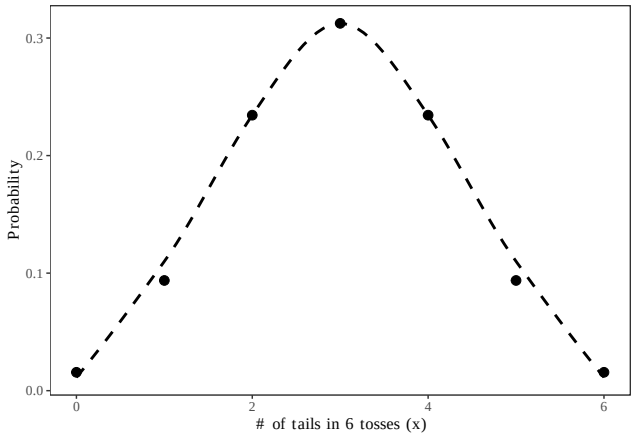

Figure 1.2: The binomial probability mass function for the 6 trial coin toss experiment

Table 1.2: P(X = x) for six tosses

x	y
0	0.016
1	0.094
2	0.234
3	0.312
4	0.234
5	0.094
6	0.016

Example 1.1. The expected value of $X \sim B(6, 0.5)$,

$$\begin{aligned} E(X) &= \sum_x x f(x) \\ &= (0 \times 0.16) + (1 \times 0.094) + (2 \times 0.234) + (3 \times 0.312) \\ &\quad + (4 \times 0.234) + (5 \times 0.094) + (6 \times 0.016) \\ &= 3 \end{aligned}$$

Importantly, while in this case $E(X)$ corresponds well to the "peak" in the PMF, this should not be assumed to be the case unilaterally, as the same expectation would result from any distribution symmetrical about $x = 3$.

1.5 Conditional Probability

Conditional probability provides a way to model the probability that an event occurs, given that another event is known to have occured. Conditional probability, as presented in Definition 1.5, requires only that the event assumed to have occured, i.e., the event we are *conditioning on*, has a nonzero probability of occuring.

Definition 1.5 (Conditional Probability)**.** *Assuming* $P(B) > 0$*,*

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

Conditional probability asserts that the probability of event A occuring given that event B occurs is equivalent to the probability of both A and B occuring (denoted as $A \cap B$) divided by the probability that B occurs. $P(A|B)$ is not generally equal to $P(B|A)$ For example, the probability that I am swimming given that I am in the water is clearly not the same as the probability that I am in the water given that I am swimming. Example 1.2 explains how to use conditional probability to calculate the probability of rolling a 2 on a 6-sided dice, given I know the outcome of the roll is less than 4.

Example 1.2. Conditional probability can be used to determine the probability of rolling a 2 on a 6 sided dice, given that I know the outcome will be less than 4. This scenario can be represented as:

$$A = \text{rolls } 2,\ P(A) = 1/6$$

$$B = \text{rolls} < 4,\ P(B) = 4/6$$

$$\begin{aligned} P(2|<4) = P(A|B) &= \frac{P(A \cap B)}{P(B)} \\ &= \frac{\frac{1}{6}\frac{4}{6}}{\frac{4}{6}} \\ &= \frac{1}{6} \end{aligned}$$

1.6 Bayes' Theorem for Discrete Values

Consider a student, Alice, who was exposed to someone with COVID-19. Being a responsible person, Alice decides she should get tested. She receives a test and the accompanying information sheet states the test is 85% accurate, meaning that 85% of the time it gives positive results to recipients who are actually positive. The sheet also says that the test yields a false positive 30% of the time, meaning that if Alice is actually negative she will still receive a positive test 30% of the time. The following day Alice receives a positive test. As a student of probability, Alice recognizes that the 85% accuracy statistic only means the conditional probability that she receives a positive test given she is COVID positive ($P(+test|covid+)$) is 85%. However, Alice is actually interested in the conditional probability that she is positive given she just tested positive, $P(covid+|test+)$. As stated in the previous section's discussion of conditional probability, $P(A|B) \neq P(B|A)$. Bayes' theorem, which follows intuitively from the theorem of conditional probability, provides this answer for Alice. We can rewrite Definition 1.5 as

$$P(A \cap B) = P(A|B)P(B).$$

Furthuremore, it can also be stated that

$$P(B \cap A) = P(B|A)P(A)$$

Clearly $P(A \cap B) = P(B \cap A)$, as both terms can be used interchangeably to represent the intersection of two sets A and B. This equivalency means that we can rewrite these equations as

$$P(B|A)P(A) = P(A|B)P(B)$$

Dividing both sides of this equivalency by $P(B)$ yields Bayes' theorem, as formalized in Definition 1.6 (Junker, 2003).

Definition 1.6 (Bayes' Theorem)**.** *Assuming* $P(B) > 0$ *and* $P(A) > 0$*,*

$$P(B|A) = \frac{P(A|B)P(B)}{P(A)}$$

Additionally, the *law of total probability*, Definition 1.7, can be used to compute $P(A)$ for a discrete sample space S_B (Wasserman, 2004).

Definition 1.7 (Law of Total Probability)**.** $B_0, ..., B_k$ *is a partition of a discrete sample space* S_B*, and*

$$P(A) = \sum_{i=1}^{k} P(A|B_i)P(B_i).$$

Returning to our example, the partition of this sample space is $[B_0 = covid+,\ B_1 = covid-]$, as Alice is either COVID positive or she is not. The final piece of information needed is the prior probability $P(B)$, which can be thought of as the likelihood of contracting covid from any given exposure. Alice did some research and concluded this likelihood is 20%. Given the information from the factsheet and Alice's prior knowledge about the probability of contracting covid, Example 1.3 shows how Alice can use Bayes' theorem to answer her question and find the probability that she is actually positive given she has tested positive.

Example 1.3. Given the following information:

$P(test+|covid+) = 0.85,\ P(test+|covid-) = 0.30,\ P(covid+) = 0.20,\ P(covid-) = 0.80$

We can represent the conditional probability that Alice is COVID positive given that she

tested positive $P(covid+|test+)$ as

$$\begin{aligned} P(covid+|test+) &= \frac{P(test+|covid+)P(covid+)}{P(test+|covid-)P(covid-) + P(test+|covid+)P(covid+)} \\ &= \frac{(0.85)(0.20)}{(0.30)(0.80) + (0.85)(0.20)} \\ &= \frac{0.17}{0.24 + 0.17} \\ &= 0.41 \end{aligned}$$

1.7 Bayes' Theorem for Probability Distributions

While Bayes' theorem can be correctly applied to discrete values, evaluating Bayes' theorem for probability distributions as an alternative to discrete values will allow uncertainty to be represented in a natural manner. In the previous example, we considered the prior probability that Alice contracts COVID in any given exposure to be an exact value of 0.20. Given that the virus that causes COVID-19 is not well understood at present moment, this value is clearly not an accurate representation of the uncertainty related to our prior knowledge as different sources may give varying values for this prior plausibility. Considering Bayes' theorem in terms of probability distributions will remove this assumption from the model in favour of a probability distribution, which represents a weighted range of all possible values of our parameter as highlighted in Definition 1.8.

Definition 1.8 (Bayes' Theorem for Probability Distributions)**.** *Assuming* $P(\theta) > 0$ *and* $P(X) > 0$*,*

$$P(\theta|X) = \frac{P(X|\theta)P(\theta)}{P(X)},$$

where $P(\theta|X)$ is the posterior probability, $P(\theta)$ is the prior probability distribution of the parameter of interest θ, $P(X|\theta)$ is the likelihood function or the probability of the data given θ, and $P(X)$ is the marginal likelihood of the data X. In this case, the marginal likelihood

$P(X)$ functions to normalize the posterior distribution and is evaluated for a continuous sample space as:

$$P(X) = \int P(X|\theta)P(\theta)d\theta$$

Chapter 2

Approximation algorithms for inference on distributions

When performing Bayesian Inference, approximation algorithms are necessary for updating the relative probability of parameters of interest in a computationally feasible manner. In modelling scenarios with multiple parameters forming a multidimensional parameter space, evaluating parameters at even a relatively small number of possible values rapidly becomes intractible. This problem is further exaggerated in the evaluation of hierarchical models, which are of particular interest in many application domains including biology, economics, chemistry, and physics. Here, we discuss markov chain monte carlo based sampling algorithms, which allow for efficiently sampling from approximations of probability distributions.

2.1 Markov Chain Monte Carlo

Markov chain monte carlo (MCMC) provides an efficient methodology for sampling from the posterior curve to perform a Bayesian update. We will first provide a theoretical background into markov chains and monte carlo approximation, followed by a discussion of *Metropolis-*

Hastings Monte Carlo, an algorithm for performing MCMC.

2.1.1 Markov Chains

A markov chain is a stochastic model expressing a sequence of possible states in which the probability of each state, X_{i-1}, depends only on the value attained in the previous state, X_{i-1}. A markov chain has several properties which are essential for its application in Bayesian statistics. Namely, each state in the chain depends only on the previous one; therefore, the markov chain preserves the assumed dependence between samples from our posterior distribution. Additionally, this localized dependence lends the markov chain another attractive feature in that it is effectively memoryless. Definition 2.1 formalizes the requirements for a markov chain (Wasserman, 2004).

Definition 2.1 (Markov Chain)**.** *A discrete sequence of random variables $X_0, X_1, ..., X_i$ is a Markov chain iff it satisfies the Markov property; that is, for all i and $x \in X$:*

$$P(X_i = x|X_0, ...X_{i-1}) = P(X_i = x|X_{i-1})$$

A markov chain is often represented as a directed graph where states in the sequence of random variables are represented as vertices, and edges represent possible paths between states whose weights are the probabilities of edges being traversed.

2.1.2 Monte Carlo Approximation

Monte Carlo approximation provides a convient method for approximately computing quantities of interest. This schema will allow for drawing samples from arbitrarily complex probability distributions. The basic example, known as monte carlo integration, evaluates an integral using monte carlo approximation (Wasserman, 2004). If we want to evaulate an

integral for some function $f(x)$, where

$$I = \int_a^b f(x)dx \ ,$$

we can approximate I using monte carlo approximation. $f(x)$ can be alternatively expressed as two functions, $h(x)$ and $w(x)$, where $h(x) = \frac{1}{a-b}$ and $w(x) = f(x)(b-a)$ as follows.

$$I = \int_a^b f(x)dx = \int_a^b w(x)h(x)dx$$

Conveniently, h is a pdf of a uniform r.v. X over $[a, b]$, which means that I can be rewritten in terms of expectation as

$$I = E_f(w(X)) \ .$$

In conjunction with the law of large numbers, this means that if we generate a sequence of random variables from a uniform distribution $X_0, ..., X_N \sim unif(a, b)$ then the standard Monte Carlo integration method asserts

$$\hat{I} \equiv \frac{1}{N} \sum_{i=0}^{N} w(X_i) \rightarrow E(w(X)) = I \ .$$

In other words, $\hat{I}$ approaches I as N grows sufficiently large. Constructing a Markov chain for evaluation with Monte Carlo will maintain the inherent, presumed dependence between samples of a sequence of random variables while facilitating the random sampling of extremely complex probability distributions. As we will expand on in the following sections, this algorithmic framework is particularly useful because it will work for models with non-normal posteriors, including hierarchical models.

2.2 Metropolis-Hastings MCMC

Metropolis-Hastings MCMC is one common algorithm for sampling from the posterior distribution to perform a Bayesian update. While it is not always the fastest in practice and requires manual tuning to work effectively, Metropolis-Hastings provides a foundation for more complex algorithms such as *Hamiltonian Monte Carlo* with *No-U-Turn sampling*. To reiterate, the purpose of sampling in Bayesian inference is to draw from some density for parameters theta. In this case, the density we are drawing from is the Bayesian posterior $P(\theta|data)$, which for the purposes of MCMC is referred to as the *target distribution*. Metropolis-Hastings algorithm provides a method for sampling from an approximation of the target distribution known as the *stationary distribution*. Critically, the law of large numbers guarantees that with sufficiently many iterations of Metropolis-Hastings, the stationary distribution will converge on the target distribution and samples drawn from the stationary distribution will appear to be samples from the target distribution. Essentially, given some sequence of states $X_0, X_1, ..., X_i$ from a Markov-Chain, where X_0 is chosen arbitrarily, an iteration of Metropolis-Hastings produces the next state to include in the sequence. As presented formally in Definition 2.2, this is achieved by generating a *proposal* for X_{i+1} from the *proposal distribution* $q(y|X_i)$, transition kernel, and then accepting the proposal with some probability dependent on the relative, target probabilities of the current state X_i and the proposal (Wasserman, 2004).

Definition 2.2 (Metropolis-Hastings MCMC)**.** *X_{i+1} is generated given $X_0, X_1, ..., X_i$ in the following manner:*

1. Sample a proposal Y from the proposal distribution $Y \sim q(y|X_i)$.

2. Evaulate the ratio of probabilities in the stationary distribution $h(x)$ for $r(X_i, Y)$ where

$$r(x, y) = min\{\frac{f(y)}{f(x)}\frac{q(x|y)}{q(y|x)}, 1\} \ .$$

3. Compute the next state X_{i+1}, where

$$X_{i+1} = \begin{cases} Y & \text{with probability } r \\ X_i & \text{with probability } 1-r \end{cases} .$$

A simplification of the acceptance probability $r(x, y)$ occurs in a special case of Metropolis-Hastings known as random walk Metropolis-Hastings, where the proposal distribution is a standard normal distribution $N(0, 1)$. This case is a random walk because the proposal Y is generated by adding a random number sampled from a standard normal distribution to X_i. When the proposal distribution is a symmetric distribution—as is the case with the normal distribution—$q(y|x) = q(x|y) \therefore \frac{q(x|y)}{q(y|x)} = 1$. This means that the acceptance probability can be simplified as follows.

$$r(x, y) = min\{\frac{f(y)}{f(x)}, 1\}$$

While Metropolis-Hastings may seem appealing because it is both memoryless and able to approximate very complex distributions, its downfall lies in the manual tuning and iteration necessary to achieve convergence of the stationary distribution to the target distribution in a reasonable number of iterations. Recall that Metropolis-Hastings generates proposals by sampling from some distribution, and in the case of random walk metropolis this is a normal distribution centered about X_i. Here, the standard deviation of the normal distribution, known as the step size, dictates the relative distance in the sample space between X_i and the generated proposal. If the step size is too low, the algorithm will make very small steps and may miss key features of the target distribution, causing the stationary distribution to require a much larger number of iterations to converge on the target distribution. On the other hand if the step size is too large, proposals will be overwhelmingly generated from the low-probability tails of the distribution, again resulting in a lack of convergence and poorly representative samples from the stationary distributon. While some methods have

been proposed for automatically tuning the step size parameter (Graves, 2011), iterating on complex, Bayesian models to tune a parameter is not particularly efficient nor computationally feasible in many cases. For this reason, other approximation algorithms are used in practical implementations of Bayesian inference. One such algorithm, *Hamiltonian Monte Carlo* (HMC), relies on theorhetical physics to compute a latent momentum variable which is applied to a hamiltonian, effectively simulating a ball rolling around the multi-dimensional sample space (Brooks et al., 2011). Despite the increased computational cost, this method is appealing because of its ability to generate proposals from distant regions of the stationary distribution with high acceptance probabilities. This means that in practice, the HMC algorithm's stationary distribution converges on the target distribution with far fewer iterations than traditional Metropolis-Hastings implementations. Though HMC still requires the user to specify a step size as well as a number of steps to move the hamiltonian before considering a proposal, in practice it is much more efficient and requires fewer iterations on a model. Additionally, a proposed extension to HMC called the *No-U-Turn Sampler* (NUTS) automatically determines the number of steps and was empirically demonstrated to perform at least equally as well as standard HMC (Homan and Gelman, 2014). The approximation algorithms breifly covered in this chapter underpin modern applications for Bayesian inference, including STAN, which are discussed in the following chapter.

Chapter 3

Modern Bayesian inference with STAN

Chapter 4

Bayesian applications in Biology: Hybrid species distribution modelling with gaussian processes

Bibliography

Brooks, S., Gelman, A., Jones, G., and Meng, X.-L. (2011). *Handbook of Markov Chain Monte Carlo*. Chapman & Hall/CRC.

Graves, T. (2011). Automatic step size selection in random walk metropolis algorithms. *Statistical Sciences*.

Homan, M. and Gelman, A. (2014). The no-u-turn sampler: adaptively setting path lengths in hamiltonian monte carlo. *The Journal of Machine Learning Research*, 15.

Junker, B. (2003). Basics of bayesian statistics.

Wasserman, L. (2004). *All of Statistics*. Springer.