Maximum likelihood

Fredrik Ronquist

September 28, 2005

1 Introduction

Now that we have explored a number of evolutionary models, ranging from simple to complex, let us examine how we can use them in statistical inference. The standard approach is to use maximum likelihood, which will be covered in this lecture.

Assume that we have some data D and a model M of the process that generated the data. The model has some parameters θ, the specific value of which we do not know but wish to estimate. If the model is properly constructed, we will be able to calculate the probability of it generating the observed data given a specific set of parameter values, $P(D|\theta, M)$. Often, the conditioning on the model is suppressed in the notation, in which case the probability would simply be written as $P(D|\theta)$. This probability is often referred to as the *likelihood* of the parameter values. In maximum likelihood inference, we simply estimate the unknowns in θ by finding the values with the maximum likelihood or, more precisely, the highest probability of generating the observed data.

2 One-parameter models

The maximization process is typically straight-forward when there is only one parameter in the model. Say, for instance, that we are interested in estimating the probability of obtaining heads when tossing a coin. A reasonable model is that each toss is identical and independent and has a heads probability of p. Hence, the probability of tails would be $1 - p$, and the probability of a particular outcome would follow a binomial distribution. For instance, the probability of the sequence HHTTHTHHTTT would be

$$L = P(D|p) = pp(1-p)(1-p)p(1-p)pp(1-p)(1-p)(1-p) = p^5(1-p)^6$$

As you can see, it is sufficient to know the number of heads and tails to calculate the probability. If we graph the likelihood we can simply find its maximum, which is at $p = 5/11 = 0.454545...$ (Fig. 1); the estimate of p is often denoted $\hat{p}$. We can also calculate $\hat{p}$ analytically by taking the derivative of L with respect to p:

$$\frac{\mathrm{d}L}{\mathrm{d}p} = 5p^4(1-p)^6 - 6p^5(1-p)^5$$

and finding where this derivative is zero. That is easily done by factoring out p^4 and $(1-p)^5$ and concentrating on the remaining expression $5(1-p) - 6p$, which will give us the only relevant root $p = 5/11$.

Figure 1: Likelihood curve for the coin tossing example. Note that the likelihood does not integrate to 1; it is not a probability distribution. The maximum likelihood estimate for p is found by locating the peak of the curve.

It is often easier to maximize the logarithm of the likelihood than the likelihood itself. In this case, we would get:

$$\ln L = 5 \ln p + 6 \ln(1 - p)$$

and the derivative

$$\frac{\mathrm{d}(\ln L)}{\mathrm{d}p} = \frac{5}{p} - \frac{6}{1 - p}$$

If we set the derivative to zero, we would get

$$\begin{aligned} \frac{5}{p} - \frac{6}{1 - p} &= 0 \\ 5(1 - p) - 6p &= 0 \\ 5 - 11p &= 0 \\ p &= \frac{5}{11} \end{aligned}$$

which, not surprisingly, is the same estimate of p. Generally in this type of model, the maximum likelihood estimate turns out to be the fraction of tosses that are heads, an intuitively appealing estimate of the probability of obtaining heads.

3 Multi-parameter models

When the model has more than one parameter it becomes much more difficult to maximize likelihood. A multi-parameter model typically has one parameter or a small set of parameters that we are interested in and a number of other parameters that are introduced in the model only to make the latter more realistic. In likelihood inference we often refer to the former parameter(s) as *structural parameter(s)* and the latter as *nuisance parameters*.

Assume that we have a parameter vector $\theta = \{\theta_1, \theta_2\}$, and that we are interested in inferring the value of the first parameter (θ_1, the structural parameter) but not the second (θ_2, the nuisance parameter). We can take two different approaches. One is to calculate the *profile likelihood*

$$L_p(\theta_1) = \max_{\theta_2} L(\theta_1, \theta_2)$$

which simply finds the likelihood of each value of θ_1 by maximizing over the possible values of θ_2.

The alternative approach is to use *integrated likelihood*, which is the same thing as marginal likelihood, although the latter term is more commonly used in Bayesian inference. With this method, the likelihood for the structural parameter is obtained by summing or integrating out the nuisance parameter:

$$L_i(\theta_1) = \sum_{\theta_2} L(\theta_1, \theta_2)$$

for a discrete nuisance parameter and

$$L_i(\theta_1) = \int_{\theta_2} L(\theta_1, \theta_2)\, \mathrm{d}\theta_2$$

for a continuous nuisance parameter.

It is easy to understand the difference between the two approaches by referring to a simple likelihood landscape for two parameters (Fig. 2). The base of the landscape is determined by the two parameters in the model and the height is the likelihood for each combination of parameter values. The profile likelihood method would obtain the likelihood curve for the parameter of interest by finding the profile of the landscape in the relevant plane, that is, its maximum height for each value of the structural parameter. The likelihood curve obtained with the integrated likelihood method would instead measure the average height of the landscape for each value of the structural parameter.

Generally speaking, there are good reasons to minimize the number of parameters that we are trying to maximize during a maximum likelihood analysis. Therefore, integration or summation is often favored where it is possible.

An interesting property of profile likelihoods is that they are scale-independent. That is, regardless of the parametrization we use for the nuisance parameter(s), the maximum of the profile likelihood for the structural parameter will be the same. However, this is not the case for integrated likelihoods. The result of integration will be dependent on how much weight we put on different parts of the nuisance parameter space. Hard-core likelihoodists will only use integrated likelihoods when there is one parametrization that is so obvious that people forget there are other possibilities, and they rarely point out that there is an assumed prior. In principle, however, integrated likelihoods always require the specification of priors. When there is an obvious parametrization, we are implicitly assuming a uniform prior on the chosen parameter space.

4 Consistency and efficiency

Generally speaking, likelihood methods have the properties of consistency and efficiency. Consistency means that the maximum likelihood estimate converges to the correct value as data accumu-

Figure 2: A likelihood landscape for a model with two parameters. We are interested in the likelihood curve for the parameter along the x axis. It can be determined using either the profile likelihood or the integrated likelihood method.

lates. Efficiency means that the variance of the estimate around the true parameter value is small and decreases rapidly with increasing amounts of data.

However, there are situations when likelihood is misbehaved. A typical situation is when the number of parameters that we are trying to estimate grows with the addition of new data; if the amount of data per parameter we are estimating is not growing, then there will always be some residual variance of the estimate that will never disappear.

Other instances occur when the model is incorrect. For instance, phylogenetic inference under over-simplified models is well known to suffer from the problem of long branch attraction. In such cases we will tend to infer ancestral states incorrectly, especially across long branches, leading to a tendency to place these incorrectly in the tree. The phenomenon is more pronounced for parsimony methods but does affect likelihood methods as well.

5 Deriving transition probabilities

To calculate the likelihood of a phylogenetic model including a tree and a substitution model, we need first to derive the transition probabilities for our substitution model. We have seen before that, if we have the instantaneous rate matrix Q, we can obtain the transition probabilities $P(t)$ over some time period t by exponentiating the Q matrix:

$$P = e^{Qt}$$

As we have discussed earlier, we can typically not estimate the absolute rate of change but only the amount of change (v), which is the product of rate and time. Therefore, we typically measure the branch length in phylogenetic trees in terms of the amount of change. In particular, we want a branch length unit to correspond to one expected change per site at stationarity. This requires a very specific scaling of the Q matrix, namely that

$$\sum \pi_i q_{ii} = 1$$

For illustration, we can scale the rate matrix for the Jukes Cantor model. The unscaled version would be

$$Q = \left\{ \begin{matrix} -3 & 1 & 1 & 1 \\ 1 & -3 & 1 & 1 \\ 1 & 1 & -3 & 1 \\ 1 & 1 & 1 & -3 \end{matrix} \right\}$$

or something similar. If we scale it with the factor μ we get

$$Q = \left\{ \begin{matrix} -3\mu & \mu & \mu & \mu \\ \mu & -3\mu & \mu & \mu \\ \mu & \mu & -3\mu & \mu \\ \mu & \mu & \mu & -3\mu \end{matrix} \right\}$$

and finally we solve for the value of the scaling factor by using the scaling condition above. Specifically, since the stationary state frequency of all nucleotides is the same (1/4) in the Jukes Cantor model, we get

$$\begin{aligned}\sum_{i=0}^{4}\pi_i(-3\mu) &= 1\\ 4\frac{-3\mu}{4} &= 1\\ -3\mu &= 1\\ \mu &= \frac{1}{3}\end{aligned}$$

If we now insert this in the Q matrix, we get the scaled version

$$Q = \left\{\begin{array}{cccc} -1 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3}\\ \frac{1}{3} & -1 & \frac{1}{3} & \frac{1}{3}\\ \frac{1}{3} & \frac{1}{3} & -1 & \frac{1}{3}\\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & -1 \end{array}\right\}$$

The principle is the same for more complex rate matrices even though the analytical expression for the scaler will be more complicated. However, it is easy enough to devise an algorithm to determine its value numerically.

To exponentiate the scaled Q matrix, we would usually find its eigenvalues and eigenvectors, decomposing it into the product:

$$Q = S\Lambda S^{-}1$$

where S is a matrix whose columns are the right eigenvectors of Q and Λ is a diagonal matrix containing the eigenvalues (λ_i). The exponential of Q can now be found by simply exponentiating each of the elements in Λ separately.

For simple instantaneous rate matrices, the eigenvalues and eigenvectors are known in closed form so that we can express the elements of the transition probability matrix exactly. For instance, the transition probability matrix for the Jukes Cantor model is

$$P = \left\{\begin{array}{cccc} \frac{3}{4}+\frac{1}{4}e^{-4v/3} & \frac{1}{4}-\frac{1}{4}e^{-4v/3} & \frac{1}{4}-\frac{1}{4}e^{-4v/3} & \frac{1}{4}-\frac{1}{4}e^{-4v/3}\\ \frac{1}{4}-\frac{1}{4}e^{-4v/3} & \frac{3}{4}+\frac{1}{4}e^{-4v/3} & \frac{1}{4}-\frac{1}{4}e^{-4v/3} & \frac{1}{4}-\frac{1}{4}e^{-4v/3}\\ \frac{1}{4}-\frac{1}{4}e^{-4v/3} & \frac{1}{4}-\frac{1}{4}e^{-4v/3} & \frac{3}{4}+\frac{1}{4}e^{-4v/3} & \frac{1}{4}-\frac{1}{4}e^{-4v/3}\\ \frac{1}{4}-\frac{1}{4}e^{-4v/3} & \frac{1}{4}-\frac{1}{4}e^{-4v/3} & \frac{1}{4}-\frac{1}{4}e^{-4v/3} & \frac{3}{4}+\frac{1}{4}e^{-4v/3} \end{array}\right\}$$

The transition probability matrix is known in closed form for many of the other simple substitution models but not for the more complex ones. In the latter cases it is common to use routines for numerically determining the eigenvalues and eigenvectors and then using those in deriving the transition probabilities. However, the eigenvalue decomposition methods can occasionally be numerically inaccurate, in which case Golub and van Loan (1996) recommend using the Padé approximation.

6 Calculating likelihoods on trees

Calculating likelihoods on trees is relatively straightforward. We assume for now that the tree and the branch lengths are given, as well as the values of all other parameters in our phylogenetic model. Typically, we are interested in calculating the total probability of the tree by summing over all possible ancestral state assignments. The summation over ancestral states is an example of the use of integrated likelihood to eliminate a nuisance parameter; this particular parameter is important to eliminate because if we tried to estimate the ancestral states (maximize over the possible assignments), we would run into the problem with the growing number of parameters that was mentioned above.

The calculation proceeds from the tip down to the root of the tree in a downpass that is very similar to the Sankoff downpass, except that we replace addition with multiplication and maximization with summation. The initialization is also slightly different. At each node in the tree we calculate the conditional probability of the tree given each of the possible state assignments at that node. As you probably recognize, this is an example of a dynamic programming algorithm.

We describe the algorithm here for a four-state DNA character, and we assume that the P matrix has been calculated already for all branches in the tree. We assign to each node p a set $G_p = \{g_A^{(p)}, g_C^{(p)}, g_G^{(p)}, g_T^{(p)}\}$ containing the conditional probability of the subtree rooted at p given each of the possible state assignments at that node. We use another similar set H_p, which will give the conditional probability of assigning each state to the ancestral end of the branch having p as its descendant. The elements of H_p will be derived from G_p and the elements of P_p, the transition probabilities for the branch ending in the node p. Initialization is performed by going through all tips and setting $g_i^{(p)} = 1$ if the state i has been observed at tip p and $g_i^{(p)} = 0$ otherwise. As usual, the downpass algorithm is formulated for a node p and its two descendant nodes q and r (Algorithm 1). At the root node ρ, we obtain the total probability of the tree by summing over all possible states, weighting each assignment by its stationary state frequency. Thus

$$L_i = \sum_i \pi_i g_i^{(\rho)}$$

Once we have the likelihood for each site we obtain the total likelihood for the entire data matrix by simply multiplying over all characters:

$$L = P(D|\theta) = \prod_i L_i$$

The entire procedure is illustrated in Figure 3.

Algorithm 1 Likelihood downpass algorithm

```
for all $i$ do
  $h_i^{(q)} \leftarrow 0$
  for all $j$ do
    $h_i^{(q)} \leftarrow h_i^{(q)} + p_{ij}^{(q)} g_j^{(q)}$
  end for
end for
for all $i$ do
  $h_i^{(r)} \leftarrow 0$
  for all $j$ do
    $h_i^{(r)} \leftarrow h_i^{(r)} + p_{ij}^{(r)} g_j^{(r)}$
  end for
end for
for all $i$ do
  $g_i^{(p)} \leftarrow h_i^{(q)} h_i^{(r)}$
end for
```

6.1 Study Questions

1. What is the difference between a profile likelihood and an integrated likelihood?

2. Why does integrated likelihoods require a prior, at least implicitly?

3. Are profile likelihoods scale independent? Are integrated likelihoods scale independent?

4. Why do we need to scale instantaneous rate matrices? How is it done?

5. Formulate the forward algorithm of an HMM by modifying the downpass algorithm for calculating tree probabilities. Tip: assume that there is only one descendant of each node in the tree and set $G_p = H_q$.

Figure 3: A worked example for the calculation of the probability of a tree. We start with transition probability matrices for each of the branches in the tree (a) and then assign conditional probabilities to the tips of the tree (b). For each node in the tree in a postorder traversal, we then calculate the conditional probabilities of the different ancestral state assignments at the bottom end of each of the descendant branches. We then multiply these costs to get the costs for the node of interest (d). Finally, at the root node, we find the total probability of the tree by calculating the weighted sum over all possible state assignments at the root, using the stationary state frequencies as the weights (e).