Numerical project (Problem set 10): Inference on biological sequence data

BIO-369

Prof. Anne-Florence Bitbol
EPFL

This numerical project will be graded and count for 40% of your final grade. Each student should hand in their personal solution as a ***Jupyter notebook using Python 3****, by uploading it on* ***Moodle*** *on* ***May 10*** *at the latest. Please clearly label question numbers using markdown cells. Please answer all questions (including those requiring sentences and not code as an answer) in the same Jupyter notebook, using markdown cells for text. Please name your Jupyter notebook* `LASTNAME_FirstName.ipynb`.

Three problem classes (April 29, May 1 and May 6) will be dedicated to this project, and during them, you can ask questions to the teaching assistants and discuss with other students as usual. However, in the end, you must hand in your personal solution. Detected plagiarism will result in a reduction of your grade.

This numerical project contains two parts that are independent from one another.

1 Statistical dependence in protein sequence data – 8 points

In this problem, we will consider the data in the file `MALGMALK1.fasta` (and then also the similar file `MALGMALK2.fasta`). This file contains an alignment of amino acid sequences of homologs of the *Escherichia coli* proteins MALG and MALK, concatenated together. In each sequence, the first 177 amino acids correspond to a MALG homolog and the next ones correspond to a MALK homolog. Here, alignment gaps will be considered just as an extra character (i.e. exactly like the letters representing amino acids).

a) In Python, load the file `MALGMALK1.fasta` and extract all the sequences in it as an array of strings. We will not need the headers for this analysis, so you can discard them. Next, transform the array of strings you obtained into a Numpy array of integer numbers, preferably by using the following mapping: A is 0, C is 1, ... [use alphabetic order for standard amino acids represented by their one-letter code]..., Y is 19 and – is 20.

b) Calculate the entropy of each column of the alignment. Calculate the mean of the entropies over the MALG sites, and then over the MALK sites. Compare them and comment.

c) Produce a `Numpy array` that contains the mutual information between each column i in the MALG homolog and each column j in the MALK homolog, in the form of a matrix. Plot the resulting matrix in colorscale. Comment on the appearance of the matrix. Also plot the histogram of the mutual informations.

d) Perform the same analysis as above for the file `MALGMALK2.fasta` (copy and adapt the code you wrote to answer the questions above).

e) Compare the results obtained: how does the appearance of the matrix of mutual information differ between the two files? How does the mean of the mutual information values you computed differ between the two datasets? One of the two files you analyzed contains an actual pair of interacting MALG and MALK homologs in each row, while in the other one, partners were scrambled (permuted) so that each row generally does not comprise the sequences of two interacting partners. Which file is which? Justify your answer.

f) Estimate the magnitude of the finite-size effects on the entropy of a site and on the mutual information of a pair of sites in this data. Comment: compare those for entropy and for mutual information, and make a comparison to the mean values of mutual information you obtained from the two datasets above.

g) What is the biological function of MALG and MALK? Comment on the specificity of their interaction. Where do you expect the pairs of sites with high mutual information to be located? Compare to the HK-RR case: what are the main similarities and differences?

h) Imagine that you were only given the file with the scrambled MALG-MALK partners, and had access to no other data. Propose a method to reconstruct the correct pairs of partners based on your findings above. You are not asked to implement the method or write code, just to briefly propose an idea for a method.

2 Maximum likelihood inference of phylogeny – 9 points

This problem does not require preexisting knowledge about phylogeny inference. All relevant notions are introduced below.

Independent site assumption. In all what follows, we will assume that each nucleotide site in a nucleotide sequence evolves independently from other sites. This assumption is a simplification – there are statistical dependencies between sites, as seen in the problem above. It is made in the vast majority of phylogeny inference approaches because it simplifies calculations.

Evolutionary distance between two nucleotide sequences. Consider two aligned nucleotide sequences of length L (i.e., comprising L nucleotides each) that have a different nucleotide at n sites, and the same one at the other $L-n$ sites. Assuming that one sequence has evolved from the other (or that both have evolved from a common ancestor), we would like to infer the evolutionary distance between them. The evolutionary distance is the average number of substitutions, i.e. mutations that change the nucleotide, that have occurred per site. It can differ from the naive estimate n/L because double substitutions can happen, e.g. $A \to C \to A$.

A few words about the Jukes-Cantor model. The Jukes-Cantor model is the simplest model of nucleotide evolution. It assumes that all substitutions ($A \to C$, $A \to G$, etc.) have the same rate and that all nucleotides are equally likely at stationary state. Let us consider a given site, which starts with the nucleotide A at the initial state. Then the sequence evolves and this site can mutate. In the Jukes-Cantor model of nucleotide evolution, after some evolutionary distance d, representing the average number of substitutions that occurred per site, the probability of having one specific nucleotide that differs from the ancestral state A at the site of interest reads:

$$p(d) = P_C(d) = P_G(d) = P_T(d) = -\frac{1}{4}e^{-4d/3} + \frac{1}{4}\,. \tag{1}$$

Meanwhile, the probability that the nucleotide is A, as in the ancestral state, reads:

$$P_A(d) = \frac{3}{4}e^{-4d/3} + \frac{1}{4} = 1 - 3p(d)\,. \tag{2}$$

These equations satisfy the normalization relation $P_A(t)+P_C(t)+P_G(t)+P_T(t) = 1$. Note that when $d \to 0$, we have $P_A(d) \to 1$ and $P_C(d) = P_G(d) = P_T(d) \to 0$, which is consistent with the fact that we start with the nucleotide A. Furthermore, when $d \to \infty$, we have $P_A(d) \to 1/4$, and the same is true for $P_C(d)$, $P_G(d)$ and $P_T(d)$. This means that under this model, if we wait long enough, we have the same probability (uniform probability distribution) to find any nucleotide at the site of interest.

More generally, $3p(d)$ represents the probability that the state of the site of interest differs between two sequences separated by an evolutionary distance d, with $p(d)$ giving the probability that it is one specific nucleotide among the 3 nucleotides that differ from the ancestral one. Meanwhile, $1-3p(d)$ represents the probability that the state of the site of interest is the same in two sequences separated by an evolutionary distance d.

a) Consider that we have data that comprises two aligned nucleotide sequences with n differences out of L sites. These two sequences are homologous and have evolved from a common ancestral sequence. For simplicity, you can consider that one is the ancestor of the others – under the Jukes-Cantor model, it is in fact equivalent. Assume that these two sequences are separated by an evolutionary distance d. What is the likelihood of such data (i.e. of data where any n sites among L differ between the two sequences) under the Jukes-Cantor model, assuming that all sites evolve independently? Express it as as a function of $p(d)$, n and L. Give the name of the probability distribution involved.

b) Now we will look for a maximum likelihood estimate of the evolutionary distance d. Explain how it can be found. Perform the corresponding calculation and show that it gives:

$$d = -\frac{3}{4}\ln\left(1 - \frac{4}{3}\frac{n}{L}\right) . \tag{3}$$

In what follows, we will denote by d_{ij} the evolutionary distance between two sequences i and j.

c) The file `BetaLactamase.fasta` contains three aligned nucleotide sequences coding for portions of beta-lactamase proteins from three different bacteria. Load the three sequences and convert letters to numbers using the mapping where A is 0, C is 1, G is 2 and T is 3. Compute the maximum likelihood estimate of the evolutionary distance d between each pair of sequences in this dataset. Denoting the sequences by 0, 1 and 2 according to their order in the file, you should thus obtain three estimated distances d_{01}, d_{02} and d_{12}.

d) We will now look at how we can infer a phylogenetic tree for these three sequences, using maximum likelihood. We will consider the three possible trees shown in Fig. 1. Which tree do you think will best describe the data? Justify your answer.

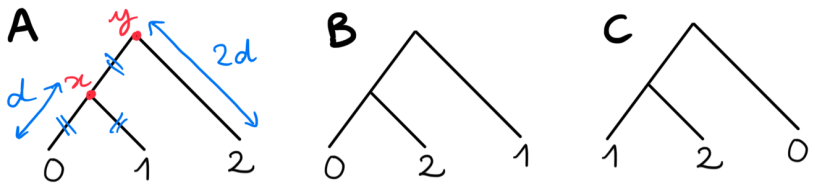

Figure 1: **Trees considered.** Each leaf of the tree corresponds to one of the observed sequences, labeled 0, 1 and 2 according to their order in the data file. In tree A, the internal nodes x and y are shown. The states of the nucleotides at these internal nodes are unknown – they correspond to ancestral states. For simplicity, we consider in all cases that branches have lengths d and $2d$, as shown on tree A. Branch lengths correspond to evolutionary distances, which will be understood in the Jukes-Cantor model here. In tree A, sequences 0 and 1 are more closely related and 2 is more distant. In tree B, 0 and 2 are more closely related. In tree C, 1 and 2 are more closely related.

e) Given the independent site assumption (see above), we will first focus on the first site of each of our three sequences. Let us call it $N_i^{(1)}$ for sequence i. We will associate to each branch of the tree the probability of the associated evolution under the Jukes-Cantor model (see above). Consider tree A in Fig. 1. Let us denote by $P\left(N_0^{(1)}\middle|N_x^{(1)}, d\right)$ the probability that we have the nucleotide $N_0^{(1)}$ at the first site in sequence 0, given that we had the nucleotide $N_x^{(1)}$ at the first site of the ancestral sequence x, and that the branch length between the two sequences is d. What is the value of this probability if $N_x^{(1)}$ is identical to $N_0^{(1)}$? What is it if $N_x^{(1)}$ differs from $N_0^{(1)}$?

f) Let us admit that the likelihood of the data observed at one site under a tree can be written as a sum over ancestral states of a product of the probabilities associated to each branch of the tree, multiplied by 1/4. Thus, the likelihood of the data observed at the first site under tree A in Fig. 1 reads:

$$\mathcal{L}^{(1)} = \frac{1}{4}\sum_{N_x}\sum_{N_y} P\left(N_0^{(1)}\middle|N_x^{(1)}, d\right) P\left(N_1^{(1)}\middle|N_x^{(1)}, d\right) P\left(N_x^{(1)}\middle|N_y^{(1)}, d\right) P\left(N_2^{(1)}\middle|N_y^{(1)}, 2d\right), \tag{4}$$

where the sums over N_x and N_y each run over all four possible nucleotides. Write a code to calculate each of the 16 terms that are summed in Eq. 4, and then the likelihood $\mathcal{L}^{(1)}$. Make the computations assuming that $d = 0.01$. What term, with what states in ancestral sites x and y, is then the largest among the 16 terms of the sum? Why?

g) Now we would like to use the full sequences and not just the first site. Given the independent site assumption (see above), how can the likelihood of the full dataset under a tree be written as a function of the likelihood of each site under that tree? Write your answer denoting by $\mathcal{L}^{(i)}$ the likelihood associated to site i (in Eq. 4, the likelihood associated to site 1 was denoted by $\mathcal{L}^{(1)}$).

h) Using the code you wrote for question f), write a code that calculates the log-likelihood (i.e. the logarithm of the full likelihood) of the full dataset under tree A in Fig. 1. The code should output a list of the values of the log-likelihoods $\ln\left(\mathcal{L}^{(i)}\right)$ obtained for each site i. Then it should calculate the total log-likelihood from them. Make the computation for $d = 0.2$. Comment on your result: Is the value of the likelihood large or small? What is the interest of taking logarithms?

i) Copy and adapt your code to compute the log-likelihood of the data under tree B, and make the computation for $d = 0.2$. The, again copy and adapt your code to compute the log-likelihood of the data under tree C, and make the computation for $d = 0.2$. Conclude: among the three trees shown in Fig. 1, what tree best describes the data according to the maximum likelihood approach? How strongly is this conclusion supported?

j) So far, we have assumed that $d = 0.2$. However, for a given tree, we can also estimate the best branch length d using maximum likelihood. Focusing on tree A, compute the total log-likelihood for the following values of d: 0.01, 0.02, 0.05, 0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1, and plot the log-likelihood versus d. Comment on your plot.