ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

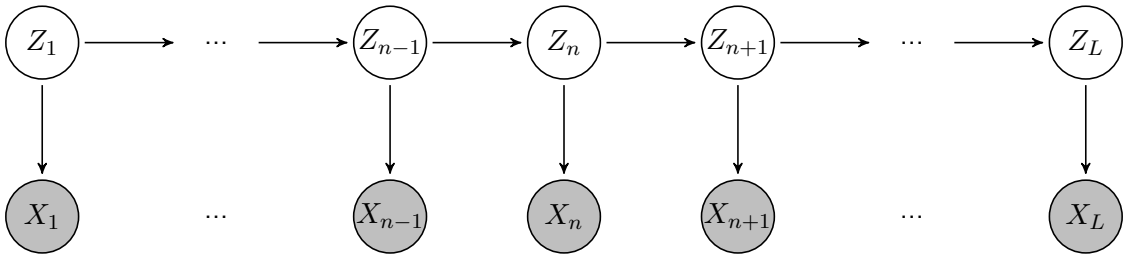
D-BSSE
Department of Biosystems
Science and Engineering

Statistical Models in Computational Biology

Jack Kuipers
David Dreifuss
Xiang Ge Luo
Rudolf Schill

Due date: 16th March 2023

Problem 6: Hidden Markov Models (2 point)



Consider the HMM represented by the above graph, where X_n are observed variables and Z_n are hidden variables. Each hidden variable can take on K different values and the observed variables can take on M different realisations. The HMM is now parameterized as in the lecture on slide 18.

(a) What is the maximum number of free parameters to define the HMM?

(b) Assume $K = 2$ and the transition matrix is

$$T = \begin{bmatrix} 0.2 & 0.8 \\ 0.6 & 0.4 \end{bmatrix}.$$

What is the stationary distribution π?

Problem 7: Predicting protein secondary structure using HMMs (8 points)

Proteins are molecules consisting of chains of amino acids. The different types of amino acids are labeled by different letters of the alphabet. A linear sequence of amino acids is called the primary structure of a protein. The pattern of hydrogen bonds in a biopolymer of a protein is the secondary structure. The Dictionary of Protein Secondary Structure (DPSS) uses seven letters to describe the pattern of hydrogen bonds in a biopolymer. One goal is to predict the secondary structure from the amino acid sequence. In a supervised learning setting, we can use amino acid sequences and matching secondary structures to predict the secondary structure.

Consider the HMM from the previous problem. The observed state space $\mathcal{X}$ for the amino acid sequence consists of the 22 letters A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, U, V, W, X, Y. The hidden state space $\mathcal{Z}$ for the secondary structure consists of 8 letters B, C, E, G, H, I, S, T. (`https://en.wikipedia.org/wiki/Protein_secondary_structure`).

You will use the Viterbi algorithm in order to predict the secondary structure of a protein. We provided you with files consisting of amino acid and secondary structure sequences, and an R implementation of the Viterbi algorithm. Every row in the tab-separated file represents one protein,

where the first column is a protein identifier, the second column is the amino acid sequence X and the third column is the matching secondary structure of the protein Z. We also provide a file with amino acid sequences only for which we want you to predict the secondary structure.

The project consists of the following steps:

(a) Read `proteins_train.tsv`, `proteins_test.tsv` and `proteins_new.tsv` into the memory and store each in a `data.frame`. (0 points)

(b) Estimate the vector of initial state probabilities I, the matrix of transition probabilities T and the matrix for emission probabilities E by maximum likelihood. (1 point)

(c) Estimate the stationary distribution π of the Markov chain by solving the eigenvalue problem (0.5 points) and by using a brute-force approach (0.5 points). (1 point)

(d) Having estimated the parameters, i.e., the emission and transition matrices E, T and the vector of initial state probabilities I, you can predict the latent state sequence Z of a protein's amino acid sequence X using the Viterbi algorithm. Use the Viterbi algorithm provided in `viterbi.r` (carefully read the parameter description!) and iterate over each `data.frame` of `proteins_test.tsv` and `proteins_new.tsv` row by row and use the amino acid sequence to *predict* its secondary structure, which you add to the `data.frame` as a new column. Save the extended `data.frame` of `proteins_new.tsv` including the predicted secondary structure as a tsv file and hand it in together with your pdf. (1 point)

(e) Estimate confidence intervals for each parameter in I, E and T with bootstrapping. In a single bootstrap run i estimate the probabilities for I_i, E_i and T_i the same as before, but not on the original data set `proteins_train.tsv`, but on the resampled data set. I.e., sample with replacement as many rows from `proteins_train.tsv` as the original data set has. Run a thousand bootstraps and compute the empirical 95% confidence intervals for each single parameter in $\{I_i\}_i, \{E_i\}_i$ and $\{T_i\}_i$. (2 points)

(f) Use the following measure to compute the accuracy of the predicted secondary structure $P = (p_i)$ for the `data.frame` of `proteins_test.tsv` given the real secondary structure $S = (s_i)$:

$$a(P, S) = \frac{1}{L} \sum_i \begin{cases} 1 & \text{if } p_i = s_i \\ 0 & \text{if } p_i \neq s_i \end{cases}$$

with sequence length L. Compute the accuracy for every protein in your `data.frame` and store the accuracies in a `vector`. What is the accuracy of the Viterbi algorithm over all sequences (i.e. call `summary` on the vector of accuracies). (2 points)

(g) Instead of using the Viterbi algorithm, now randomly guess secondary structures for all sequences. Compare the global accuracies of the Viterbi and the random approach and plot all accuracy distributions using boxplots. (1 point)

References

[1] Durbin, Richard and Eddy, Sean R and Krogh, Anders and Mitchison, Graeme *Biological sequence analysis: probabilistic models of proteins and nucleic acids*. Cambridge university press, 1998

[2] Protein Structure. `https://en.wikipedia.org/wiki/Protein_structure`