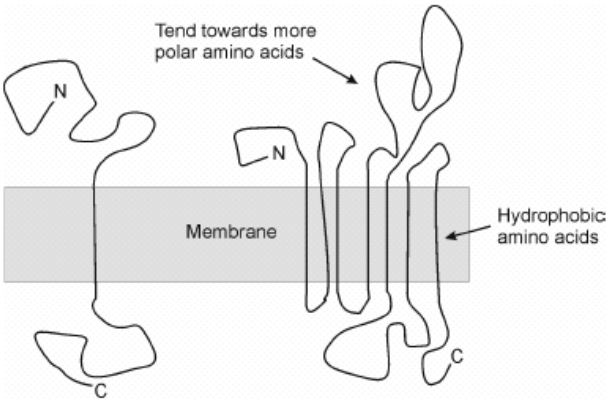
Problem Set #1 BTE4402 Total 15 points

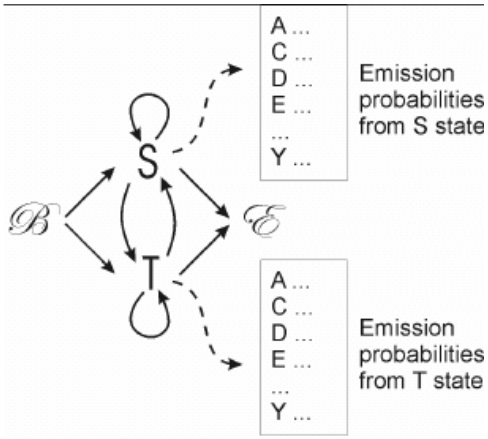
A Hidden Markov Model for finding transmembrane protein sequences

Proteins in a cell are often synthesized by a ribosome at one location in a cell (for example, in the cytosol or attached to the endoplasmic reticulum), then shipped someplace else in the cell where the proteins actually do their work. About a fourth to a third of the proteins get attached to membranes, often with a portion of the protein on one side of a membrane, a portion penetrating the membrane, and a portion on the other side of the membrane:



In this problem, we'll train a hidden Markov model to detect the regions of these so called "integral membrane proteins" that end up buried in the membrane. The reason we can do this is because the amino acids in these regions tend to be much greasier in character than the amino acids in the non-membrane-associated part of the protein.

We'll construct a very simple hidden Markov model of the following architecture:



where $\mathscr{B}$ is the beginning state, S is the hidden state for an amino acid in a water soluble segment of the protein, T is the hidden state for an amino acid in a trans-membrane segment, and $\mathscr{E}$ is the end state. In reality, we won't model the end state, so you can

essentially ignore it for this exercise. So, the protein sequence on the left in the example above would be something like SSSSSSSSSSSSSSTTTTTTTSSSSSSSSSSS and the protein sequence on the right would have seven runs of T's, flanked and interspersed with runs of S's.

From Problem_Set01.zip, download 3 txt files:
(1) soluble_sequences, (2) transmembrane_sequences, and (3) state_sequences.
These files have, respectively, a set of amino acid sequences from soluble segments of proteins, a set of amino acid sequences from trans-membrane regions of proteins, and a set of state sequences for all of the known integral membrane proteins of yeast.

(P1) Calculate the frequencies of amino acids in the first two files: soluble_sequences and transmembrane_sequences (You can write program for this job or use emissionP.py in Problem_Set01.zip). Use these frequencies to generate the emission probability tables of the S and T states of the HMM. **(2 points)**

(P2) Calculate frequency of each of the 2 states in the third file: state_sequences. Use these frequencies as the transition probabilities from the beginning state of the HMM (Start probability).
Also calculate the number of occurrences of each digram in the state file (e.g., SS, TT, ST, or TS. A digram is a 2 character string. In general, a string of *n* characters is a *n*gram.). From the numbers of observations, you should be able to construct the transition probability matrix for the HMM. (You also can write programs for this job or use transitionP.py in Problem_Set01.zip) **(2 points)**
(Notice) A transition probability from S to S = count(SS)/[count(SS)+count(ST)].

(P3) Now, take the amino acid sequence `DELIFFLIF` and calculate the most likely state sequence using the Viterbi algorithm.
Turn in the Viterbi matrix, and the most likely state sequence. **(2 points)**
You MUST show your calculation process. Or you won't get any point.
If you write a program for this decoding process, you will get extra 3 points.

More examples of HMM
(P4) Draw a reasonable topology for an HMM that would identify secondary structures in proteins. **(1 point)**

(P5) Suggest an application (ANY application!) of HMM's to a biological problem OTHER THAN the applications we have discussed already (e.g., 5' splice site recognition, CpG islands, protein trans-membrane segment prediction, or protein secondary structure prediction). Suggest a reasonable HMM topology that could be applied to the problem (**Please do not make copy of any published article**). **(1 points)**

Sequence Logo analysis

Sequence Logo is to visualize motif of the given aligned target sequences. We have discussed about two web tools to perform Sequence Logo analysis, WebLogo (https://weblogo.berkeley.edu/logo.cgi) and enoLOGOS(http://www.benoslab.pitt.edu/cgi-bin/enologos/enologos.cgi). These are problems to practice those web tools. Download "**E.coli_Arginine_Repressor_binding_sites_17.txt**" in Problem_Set01.zip. This file contains 17 aligned sequences.

(P6) Using WebLogo create and report Sequence Logos with information contents. **(1 point)**

(P7) Using enoLOGOS create Sequence Logos with relative entropy with the following different genomic GC%; (1) 50%, (2) 30%, (3) 70%. How are they different? Explain why. **(2 points)**

Phylogenetic tree construction

DnaA is a protein that activates initiation of DNA replication in bacteria. Thus, the dnaA gene plays a critical and essential role in bacteria. You will build a gene tree using dnaA homologous protein sequences of several bacterial species living in the human gut. This process will consist of three main steps:

1. Multiple sequence alignment using ClustalX

(Step 1) Install and start ClustalX (http://www.clustal.org/download/current/). Download the appropriate ClustalX installation file for your computer OS and install it. You can find a description of each file in Readme. After installation, start ClustalX. Then a window should open on your desktop.

(Step 2) Download "dnaA_homolog_sequences.fa" in Problem_Set01.zip and input this fasta file into ClustalX (Click File → Load sequences and select this fasta file). After loading, you should see all 20 dnaA protein sequences of bacterial species on the ClustalX window.
(Notice) If you get an error "Cannot open file!", Korean is in your fasta file absolute path. After moving this fasta file to a path containing only English, try again.

(Step 3) Run the alignment (Click Alignment → Do Complete Alignment). Before the alignment begins, the program will ask you for two output file names. You can click "OK" unless you want to save outputs as different file names. The alignment may take a few seconds to a few minutes to complete—progress will be displayed at the bottom of the window.

(Step 4) Once the alignment is complete, use the scroll bar to look at the overall alignment. Since each amino acid is highlighted with a different color, it's easy to see where the alignment is good. An asterix (*) is printed at the top of the alignment where all sequences

agree on a particular location, and a hyphen (-) is put in where a gap has been inserted by the program to make the alignment work well.

2. Gene tree construction using IQ-TREE

(Step 5) To construct a gene tree, open the IQ-TREE web server (http://iqtree.cibiv.univie.ac.at/). Input alignment file into IQ-TREE web server (Block "Input Data" → Row "Alignment file :" → Click "Browse…" → Select .aln file). The .aln file contains the alignment results in clustal format. After selecting the .aln file, click "SUBMIT JOB" at the very bottom of the homepage.

(Step 6) After submitting a job, you should see the new page of the IQ-TREE webserver. You can see the submitted job list on the left side of the new page, and the list may contain one job. Through the "Status" column of the list, you can know the progress of the tree construction (For progress updates, press F5 to refresh.). The tree construction may take one minute.

(Step 7) After the job status becomes "success", you can download the results by clicking the "DOWNLOAD SELECTED JOBS" in the bottom left corner. Among the result files, download the .treefile file. This file contains the dnaA gene tree in Newick format.

3. Tree visualization using iTOL

(Step 8) To visualize the gene tree, open iTOL (https://itol.embl.de/upload.cgi). iTOL is an online tool for the display, annotation, and management of phylogenetic trees. Input the treefile into iTOL (Click "파일 선택" → Select .treefile) and click "Upload". You should see the new page showing the gene tree.

(P8) Report the dnaA gene tree of 20 bacterial species created by following the steps above. You should change the following options in the basic control panel: Set "Unrooted" in "Mode" and dashed lines 3px in "Branch options". **(2 points)**

(P9) Interpret the results of multiple sequence alignment and gene tree in terms of taxonomy. **(2 points)**