Assignment 3 - Hidden Markov Models

COMP 561 - Computational Biology Methods and Research

LE, Nhat Hung
McGill ID: 260793376
Date: November 4, 2019
Due date: November 11, 2019

Prof. Mathieu Blanchette
Fall 2019

Question 1 (80 points).

In this question, you will implement the Viterbi algorithm and apply it to the gene finding HMM seen in class to predict genes in a bacterial genome.

Vibrio cholerae is the bacteria that causes cholera. It was first sequenced in 2000 (see https://www.nature.com/articles/35020000). Download the genome sequence here:

ftp://ftp.ensemblgenomes.org/pub/bacteria/release-45/fasta/bacteria_81_collection/vibrio_cholerae/dna/Vibrio_cholerae.GFC_11.dna.toplevel.fa.gz

Note: Although the genome of *Vibrio cholera* consists of two chromosomes, the sequence file contains more than two sequences, because it is incompletely assembled.

Download the gene annotation here:

ftp://ftp.ensemblgenomes.org/pub/bacteria/release-45/gff3/bacteria_81_collection/vibrio_cholerae/Vibrio_cholerae.GFC_11.37.gff3.gz

The format of both files should be fairly self-explanatory, but you can find more information about the gff3 file format here: http://gmod.org/wiki/GFF3. One important detail we've omitted in our in-class discussion is that genes can actually be located either on the forward or the reverse DNA strand. In class, we have only considered the genes located on the forward strand. In this assignment, we will also only focus on the genes on the forward strand. It is not much more complicated to handle genes on the negative strand, but it's just annoying, so we will pretend that genes on the negative strand do not
exist, i.e. that portions of the genome occupied by genes on the reverse strand are actually intergenic regions.

a) (10 points) Using the genome sequence and gene annotation for Vibrio cholerae, infer the (i) average length of intergenic regions, (ii) average length of genic region, (iii) the nucleotide frequency table for intergenic regions; (iv) the codon frequency table for genic regions.

Solution:

(i) Average length of intergenic regions: 1290

(ii) Average length of genic region: 991

(iii) Nucleotide frequency table for intergenic regions:

```
{'A': 0.266, 'C': 0.243, 'G': 0.226, 'T': 0.265}
```

(iv) Codon frequency table for genic regions:

```
{'AAA': 0.036, 'AAC': 0.02, 'AAG': 0.014, 'AAT': 0.019, 'ACA': 0.008, 'ACC': 0.021,
 'ACG': 0.011, 'ACT': 0.013, 'AGA': 0.003, 'AGC': 0.014, 'AGG': 0.001, 'AGT': 0.011,
 'ATA': 0.004, 'ATC': 0.025, 'ATG': 0.026, 'ATT': 0.031, 'CAA': 0.033, 'CAC': 0.011,
 'CAG': 0.018, 'CAT': 0.013, 'CCA': 0.013, 'CCC': 0.006, 'CCG': 0.011, 'CCT': 0.011,
 'CGA': 0.005, 'CGC': 0.018, 'CGG': 0.003, 'CGT': 0.02, 'CTA': 0.009, 'CTC': 0.014,
 'CTG': 0.029, 'CTT': 0.013, 'GAA': 0.039, 'GAC': 0.014, 'GAG': 0.024, 'GAT': 0.037,
 'GCA': 0.019, 'GCC': 0.022, 'GCG': 0.03, 'GCT': 0.021, 'GGA': 0.007, 'GGC': 0.025,
 'GGG': 0.009, 'GGT': 0.027, 'GTA': 0.011, 'GTC': 0.014, 'GTG': 0.029, 'GTT': 0.016,
 'TAA': 0.002, 'TAC': 0.014, 'TAG': 0.001, 'TAT': 0.016, 'TCA': 0.01, 'TCC': 0.006,
 'TCG': 0.009, 'TCT': 0.011, 'TGA': 0.001, 'TGC': 0.004, 'TGG': 0.013, 'TGT': 0.006,
 'TTA': 0.019, 'TTC': 0.014, 'TTG': 0.023, 'TTT': 0.026}
```

b) (30 points) Using the programming language of your choice, implement the Viterbi algorithm. Your program does not need to work with an arbitrary HMM, but only needs to work for the specific bacterial gene-finding HMM seen in class (4 states: Intergenic, Start, Middle, Stop). The program's argument should be

I. A Fasta file containing one or more bacterial genome sequences.

II. A configuration file (formatted as you want), which contains the information found in (a), to be used to determine the emission and transition probabilities for your HMM. (Assume that the initial state probability is 1 for the intergenic state, and zero for all other states). The output should be a GFF3 file with one line per predicted gene (only lines corresponding to portions annotated as genes need to be included in the output file), e.g. something like:

```
DN38.contig00011 ena CDS 85269 86705 . + 0 .
DN38.contig00011 ena CDS 87006 87230 . + 0 .
DN38.contig00011 ena CDS 88079 89323 . + 0 .
```

Submit your code, along with a simple explanation about how to run it.

Solution:

Refer to the code and instructions in `README.md`.

c) (10 points) *Vibrio vulnificus* is a species of bacteria closely related to *Vibrio cholerae*. In 2005, it caused an outbreak following the hurricane Katrina (https://www.ncbi.nlm.nih.gov/pubmed/16195696). Its genome was sequenced and can be downloaded here:

ftp://ftp.ensemblgenomes.org/pub/bacteria/release-45/fasta/bacteria_64_collection/vibrio_vulnificus/dna/Vibrio_vulnificus.ASM74310v1.dna.toplevel.fa.gz

Run your program on this genome, using the parameters obtained for *Vibrio cholerae*. Submit the GFF3 file with your gene predictions (forward strand only).

Solution:

Reproduce the output by following the instructions in `README.md`.

d) (15 points) The genes of *Vibrio vulnificus* were annotated based on a variety of types of evidence. You can download the GFF3 file here:

ftp://ftp.ensemblgenomes.org/pub/bacteria/release-45/gff3/bacteria_64_collection/vibrio_vulnificus/Vibrio_vulnificus.ASM74310v1.37.gff3.gz

Evaluate the accuracy of your predictions against the gene annotation available here. Specifically, report:

- The fraction of annotated genes that:
    - Perfectly match both ends of one of your predicted genes
    - Match the start but not the end of a predicted gene

  - Match the end but not the start of a predicted gene
  - Do not match neither the start not the end of a predicted gene
- The fraction of your predicted genes that:
  - Perfectly match both ends of one of an annotated genes
  - Match the start but not the end of an annotated gene
  - Match the end but not the start of an annotated gene
  - Do not match neither the start not the end of an annotated gene

Solution:

Annotated genes that

- Perfectly match both ends of a predicted gene: 0.86%
- Match the start but not the end of a predicted gene: 4.46%
- Match the end but not the start of a predicted gene: 80.27%
- Don't match neither the start nor end of a predicted gene: 14.41%

Predicted genes that

- Perfectly match both ends of an annotated gene: 0.59%
- Match the start but not the end of an annotated gene: 0.61%
- Match the end but not the start of an annotated gene: 57.43%
- Don't match neither the start nor end of an annotated gene: 41.36%

e) (15 points) What properties of annotated genes are associated to an elevated risk of being partially or completely missed by your predictor? What are the properties of genes predicted by your predictor that do not match an annotated gene? Write a paragraph for each question, supported by data and/or figures

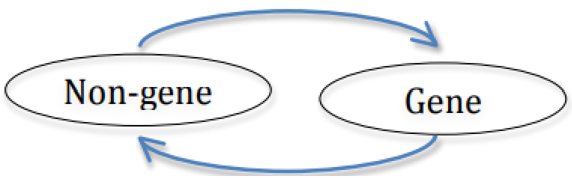
Solution:

Question 2 (20 points).

If we think of an HMM as a machine to generate a random sequence of observations, the number of consecutive steps the path will remain at a given state follows a geometric distribution (https://en.wikipedia.org/wiki/Geometric_distribution). This means, for example, that, in our gene-finding HMM, the length distribution of exons, introns, and intergenic regions will be assumed to be geometric. However, in reality, these regions have length distributions that can be far from geometric. Consider the very simple two-state gene finding HMM shown below. Assume that we have a desired gene length distribution, provided in the form of a discrete probability distribution for lengths ranging from 1 to 1000:

$$Pr[\text{length} = k] = p_k$$

Describe (in at most half a page) how you could modify the HMM below to produce a second HMM where the distribution over the duration of stay in the set of gene states is exactly the target length distribution. Note that the Gene state will probably need to be subdivided in several Gene sub-states. Describe not only the states of your HMM but also the transition probabilities.



Solution:

From the above HMM, let *p* = probability of transitioning into a genic state.

We want:

$$\text{product of probabilities up to k-th genic state} = p_k$$
$$\Rightarrow p^k = p_k$$
$$\Rightarrow p = p_k^{1/k}$$

However, the above is not feasible, because:

1. *k* varies, while we only have 1 *p*
2. While generating a sequence, we have no way of knowing *k*

Therefore, we will break the 1 genic state into 1000 genic states in order to have 1000 genic transition probabilities.

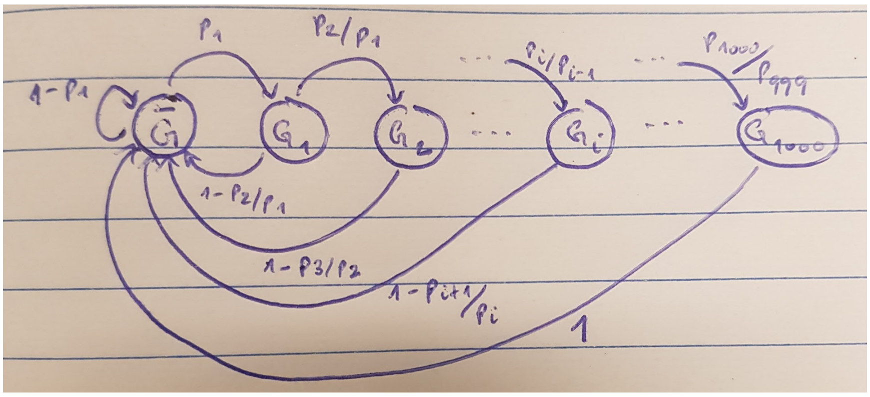
We define then:

$$G_k \text{ genic state at length k}$$
$$\overline{G} \text{ non-genic state}$$
$$T(i) \text{ probability of transitioning into genic state at length } i$$

Then, for a genic region of length k, the probability is:

$$\prod_{i}^{k} T(i) = p_k$$

The above is accomplished through the following HMM:



The transition probability matrix is:

$$T = \begin{array}{c|ccccccccc|}
 & \overline{G} & G_1 & G_2 & G_3 & \dots & G_k & G_{k+1} & \dots & G_{1000} \\
\hline
\overline{G} & 1-p_1 & p_1 & & & & & & & \\
G_1 & 1-p_2/p_1 & & p_2/p_1 & & & & & & \\
G_2 & 1-p_3/p_2 & & & p_3/p_2 & & & & & \\
\vdots & \vdots & & & & \ddots & & & & \\
G_{k-1} & 1-p_k/p_{k-1} & & & & & p_k/p_{k-1} & & & \\
G_k & 1-p_{k+1}/p_k & & & & & & p_{k+1}/p_k & & \\
\vdots & \vdots & & & & & & & \ddots & \\
G_{1000} & 1 & & & & & & & & 0 \\
\hline
\end{array}$$