Project1

Mateo Sokac

September 24, 2018

Hidden Markov Model: Project 1

Advanced Statistical Methods in Bioinformatics, Asger Hobolth

Mateo Sokac

1) Jukes - Cantor model

In the Jukes-Cantor model, the probability p that two nucleotides are different in a genomic position is given by

$$p = 1 - I(t) = 3/4(1 - exp(-4t/3))$$

where t is the evolutionary distance (or, more precisely, the expected number of substitutions) between the two genomes.
(a) Show in detail that we can invert the equation to obtain:

$$t = -\frac{3}{4}log(1 - \frac{4}{3p})$$

We start with:

$$p = 1 - I(t) = \frac{3}{4}(1 - exp(-\frac{4t}{3})) - > 3/4$$

$$4/3p = 1 - exp(-3t/4)$$

$$1 - 4/3p = exp(-4t/3) - > Ln$$

$$Ln(1 - 4/3p) = -\frac{4}{3t}$$

$$-\frac{4}{3}Ln(1 - \frac{4}{3p}) = t$$

(b) Show that t ~= p for small p (use a Taylor expansion), and

$$t - > \infty forp - > 3/4.$$

Argue why these two limitating cases make sense.
Two limiting cases make sense because if we use 0 for p,log expression will return log(1) = 0, and if we use 3/4 as p, log expression will return

$$log(0) = -\infty$$

Using Taylor expansion we use the following formula:

$$f(x) = f(x_0) + f'(x) * (x - x_0)$$

In our example we want to prove that if we use a small number for p (let's say 0) the t(p) would be approx p.

$$f(x_0) = -3/4 * log(1 - (4/3 * 0)) = 0$$

$$f'(x) = \frac{-3log(1 - 4x/3)}{4} = \frac{1}{1 - \frac{4x}{3}} = -\frac{3}{4x - 3}$$

Putting 0 in frist derivative results in:

$$f'(x_0) = -\frac{3}{4 * 0 - 3} = 1$$

And returing the values to original formula on the begining we get:

$$f(x) = f(x_0) + f'(x) * (x - x_0)$$

$$f(t) = 0 + 1 * (p - 0)$$

$$t \approx p$$

Using taylor expression for the second case we get:

$$f(x) = f(x_0) + f'(x) * (x - x_0)$$

$$f(x) = \infty + \infty * (p - 3/4)$$

$$f(x) = f(x_0) + f'(x) * (x - x_0)$$

$$f(x) = \infty$$

(c) The probability of a base pair (nucleotide) position in the human genome being different from the homologous position in the chimpanzee genome is 1.37%, and the substitution rate is 0.5 · 10 -9 substitutions per base pair per year. What is the average divergence time (in million years) between human and chimpanzee? The difference between human and gorilla is 1.75% and human and orangutan is 3.40%. What is the divergence time between human and gorilla, and human and orangutan?

$$S_{rate} = 0.5 * 10^{-9}$$

Using following function made in R we can get the following results:

```r
calcT <- function(p, s){
  # divided By substition rate, divided by milion,
  # divided by 2 because diploidity
  return(-3/4 * log(1 - (4/3*p) ) / s / 10^6 / 2)
}
```

```
## Human - Chimp:  13.82667 million years

## Human - Gorilla:  17.7074 million years

## Human - Orangutan:  34.79478 million years
```

Identifying neutral and functional regions (or: Phylogenetic footprinting)

We investigate two aligned sequences S1 and S2. The sequences are of length L = 293 after removal of gaps. The first and the last part of the sequences look as follows:

$$S1 : CCTTGG...TGAGGGT$$

$$S2 : GCCTTCG...AGAGGGT$$

We first translate the nucleotide sequences to a numerical vector. We translate A,G,C,T to 1,2,3,4. The first and the last part of the numerical sequences look as follows:

$$S1.num : 2334422...4212224$$

$$S2.num : 2334432...1212224$$

There is a total of 16 possible outcomes in each alignment column (4 possible outcomes in S1 times 4 possible outcomes in S2). We can translate the 16 possible outcomes to the numbers 1, . . . , 16 using the command

```
obs.seq <- 4*(S1.num-1)+S2.num
```

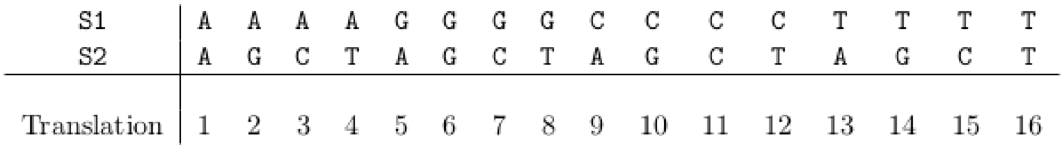
The first and last part of obs.seq look as follows:

$$6, 11, 11, 16, 6...$$

(a) Explain that a column with A in S1 and A in S2 is translated to the number 1, while a column with G in S1 and C in S2 is translated to the number 7.
Using code given in the example we can see the following

$$A - A = 4 * (1 - 1) + 1 = 1$$

$$G - C = 4 * (2 - 1) + 3 = 7$$



S1	A	A	A	A	G	G	G	G	C	C	C	C	T	T	T	T
S2	A	G	C	T	A	G	C	T	A	G	C	T	A	G	C	T
Translation	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Table 1: Translation from possible columns in the alignment to the numbers 1, . . . , 16.

(b) Explain that the following vector of length 16 indicates if an alignment column contain similar or different nucleotides:

The vector corresponds to the Table 1, where index of Table 1 indicates if nucleotides are the same or not. If they are, we vector value will be 0, if they are not vector value will be 1, so we can use it later for finding number of differences.

$$diff.indx < -c(0, 1, 1, 1, 1, 0, 1, 1, 1, 1, 0, 1, 1, 1, 1, 0)$$

From diff.indx and obs.seq we make a new vector, diff.seq, of the same length as S1 and S2, which indicates if the two nucleotides are the same or not. The R command and the first and last entries in diff.seq are. We find the total number of different columns in the alignment with the command and we get:

```
diff.indx <- c(0,1,1,1,1,0,1,1,1,1,0,1,1,1,1,0)
diff.seq <- diff.indx[obs.seq]
sum(diff.seq)
```

```
## [1] 69
```

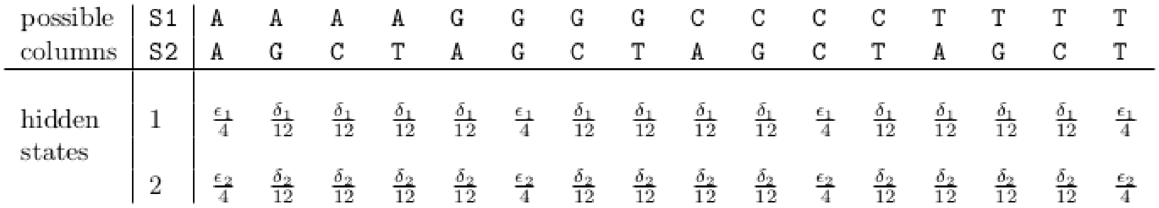
(c) Use the previous exercise to estimate the total number of expected substitutions between the two sequences under the Jukes-Cantor model.

```
p = 69/293

t <- (-3/4)*log(1-((4/3)*p))

expected_nsubs <- t * 293
```

```
## Expected number of substitutions we get is  82.81668
```

Although we only observe 69, we can see that expected number of substitions is higher, since we don not take into account consecutive substitions. For example A going to C and going back to A.

We now want to inspect the two sequences in more detail. In particular we want to divide the sequences into two types of regions: Neutrally evolving regions and functional regions. If a region is functional, the number of substitutions is expected to be smaller than if the region is evolving neutrally. In order to divide the sequences into these two types of regions, we formulate a hidden Markov chain with two hidden states. The hidden states correspond to functional regions (state 1) and neutral regions (state 2). Let delta_1 be the probability for observing a column with different nucleotides in state 1, and let delta_2 be the probability for different nucleotides in state 2. Furthermore, let i = 1 - delta_i (i = 1, 2) be the probability for observing a column in state i where the two nucleotides are the same. The emission probabilities are determined from the matrix b i (a) in Table 2, where i = 1, 2 and a = 1, . . . , 16.



possible	S1	A	A	A	A	G	G	G	G	C	C	C	C	T	T	T	T
columns	S2	A	G	C	T	A	G	C	T	A	G	C	T	A	G	C	T
hidden states	1	$\frac{\epsilon_1}{4}$	$\frac{\delta_1}{12}$	$\frac{\delta_1}{12}$	$\frac{\delta_1}{12}$	$\frac{\delta_1}{12}$	$\frac{\epsilon_1}{4}$	$\frac{\delta_1}{12}$	$\frac{\delta_1}{12}$	$\frac{\delta_1}{12}$	$\frac{\delta_1}{12}$	$\frac{\epsilon_1}{4}$	$\frac{\delta_1}{12}$	$\frac{\delta_1}{12}$	$\frac{\delta_1}{12}$	$\frac{\delta_1}{12}$	$\frac{\epsilon_1}{4}$
	2	$\frac{\epsilon_2}{4}$	$\frac{\delta_2}{12}$	$\frac{\delta_2}{12}$	$\frac{\delta_2}{12}$	$\frac{\delta_2}{12}$	$\frac{\epsilon_2}{4}$	$\frac{\delta_2}{12}$	$\frac{\delta_2}{12}$	$\frac{\delta_2}{12}$	$\frac{\delta_2}{12}$	$\frac{\epsilon_2}{4}$	$\frac{\delta_2}{12}$	$\frac{\delta_2}{12}$	$\frac{\delta_2}{12}$	$\frac{\delta_2}{12}$	$\frac{\epsilon_2}{4}$

Table 2: Emission probabilities from the two hidden states. We have $\epsilon_1 = 1 - \delta_1$ and $\epsilon_2 = 1 - \delta_2$.

(d) d. Explain why the emission probabilities sum to 1 within each row.

Because Emission probability matrix is a stochastic matrix describing probabilities of each state emitting n number of emissions. Since, it is a stochastic process, all possabilities for a single state must sum up to 1. If rows(in our example) would not sum up to 1, the emission probability matrix would be wrong.

The transition matrix is determined by the 2 × 2 matrix

$$P = \begin{pmatrix} p11 & p12 \\ p21 & p22 \end{pmatrix}$$

and the initial state is given by

$$\pi = (1/2, 1/2)$$

We want to estimate the parameters using the EM-algorithm.

(e) Show that the full likelihood:

$$L(\theta, Q, O) = \pi_{q1} b_{b1}(O_1) \prod_{t=2}^{L} (p_{qt-1,qt} b_{qt}(O_t))$$

is proportional to

$$L(\theta, Q, O) \propto p11^{N11} p12^{N12} p22^{N22} p21^{N21} \delta_1^{N_1^\delta} \delta_2^{N_2^\delta} \delta_3^{N_3^\delta} \delta_4^{N_4^\delta}$$

and describe in words the meaning of N_ij , N_i_delta and others. . .
Full liklihood for Hidden Markov Models is defined with 3 parameters, initial distribution, transition matrix and emission matrix (π, P, b). Full likelihood of those three parameters can be defined as:

$$L(\pi, P, b) = \pi_{q1} [\prod_{t=1}^{T-1} P_{qt,qt+1}] [\prod_{t=1}^{T} b_{qt}(O_t)]$$

$$= \prod_i [\pi_i^{1(q=i)}][\prod_{i,j} p_{ij}^{N_{ij}}][\prod_{ia} b(a)^{N_i(a)}]$$

This line shows us that we can use observed counts(frequencies) as our estimates. Observed counts(frequencies) can be defined as:

$$N_{ij} = \sum_{t=1}^{T-1} 1(q_t = i, q_{t+1} = j)$$

Meaning we can sum all the states where state i goes to state j, all emissions emitted from state i, and so on...
Speaking in log-likelihood approach we can write the same formula as:

$$l(\pi, p, b) = \sum_i 1(q = i)Log(\pi_i) + \sum_{ij} N_{ij}Log(p_{ij}) + \sum_{i,a} Log(b_i(a))$$

Replacing indicator function with expectations we get conditional probabilities:

$$= P(p = i|O) + E[N_{ij}|O] + E[N_i(a)|O]$$

Each expression for expectation with conditional probabilities of observed sequenced can be rewritten(or calculated) as following formula, since we can emit a in state i, which corresponds to multinomial distribution:

$$b_i(a) = \frac{E[N_i(a)|O]}{E(N_i|O)}$$

The same logic goes for P_{ij}

$$p_{ij} = \frac{E[N_{ij}|O]}{E[N_i|O]}$$

and on the end for initial distribution we know that is has to sum up to 1 and it is raised to some expression which comes from multinomial distribution, meaning we can observe frequencies. The joint probability of q = i given the observed sequence O divided by probability of sequence O, or expected proportion of times in state S_i at first time point, given sequence O.

N_{ij} expression refers to a observed count of transtion from i to j. Computing temporary transition or emission matrix boils down to count down the observed states or emission and update the transition matrix until it converges to some values (or likelihood does not change for a lot). δ_{ij} is the expression for the emission counts and it follows the same logic as for N_{ij}.

(f) Describe the M-step in the EM-algorithm (explain the maximization of (1)).

In M-step we update existing values of transition matrix or emission matrix based on observed values (counting them). This step is done before starting the new iteration, so we can use the newly created values in next iteration. Algorithm is maximizing likelihood of a model given the data in every step since it is slowly updating paramteres of a model, increasing the likelihood in every iteration. M step (maximization step), maximazes the log likelihood based on the values found in E step(estimation).

(g) Describe what type of calculations are needed for the E-step in the EM-algorithm.

Expected values are computed by following formulas:

$$\pi_i =$$

The expected proportion of times in state S_i at first time point, given O^d .

$$p_{jk} = \frac{E(N_{jk}|O^d}{E(N_i|O^d)}$$

,

$$b_i(a) = \frac{E(N_i(a)|O^d}{E(N_i|O^d)}$$

Those formulas can be interpreted as number of characters given the sequence O. Meaning we have to count the each to each state transtition and also count eash state emitting specific emission and divide by all possibilites, in both cases. The results we get would be used in M step in order to maximaze liklihood and proceed with those values in next iteration.

We now want to implement the EM-algorithm. We first implement a function, EmissionFct, which returns the matrix in Table 2 as a function of delta_1 and delta_2 . Second we activate the function forwardBackward (previously used in the course). Third we apply the EM algorithm:

(h) Explain the R program above.
MISSING ANSWER
After 20 iterations the parameter estimates are

$$\theta = (\delta_1, \delta_2, \delta_3, \delta_4) = (0.12, 0.34, 0.84, 0.86)$$

(i)What is the expected number of substitutions between the two sequences under the hidden Markov model?
MISSING ANSWER

(i) Decode the sequence alignment using (a) the Viterbi algorithm and (b) posterior decoding. Make a plot that summarizes the data and corresponding decoding. Do the regions with few differences correspond to predicted functional regions?
MISSING ANSWER