May 2014 - Bio2

1. Question 1

   You are given three possible models for a specific region of DNA:

   (a) The independent model

   (b) Derived from the nucleotide counts: $N_A = 200$, $N_C = 100$, $N_G = 100$, $N_T = 600$

   (c) Derived from the nucleotide pair counts (first nucleotide is vertical):

	A	C	G	T
A	10	80	5	5
C	35	10	10	45
G	30	20	20	30
T	60	10	25	5

   1.1 State the relevant probabilities for each model. Explain the differences between these models. What are the main advantages and disadvantages of models (b) and (c) when used for the purpose of identifying specific sections of DNA with differing statistics?

a) For the independent model, A,T, C and G have equal probabilities such that

P(A)+P(T)+P(C)+P(G) = 1

hence, P(A)=P(T)=P(C)=P(G) = ¼ = 0.25

b) For the multinomial model,

P(A) = NA/N = 200/(200+100+100+600) = 0.2 [where N = NA+NT+NC+NG]

Similarly,
P(T) = 0.6
P(C) = 0.1
P(G)= 0.1

c) What does this table represent? Does the position (A,A)=10 show that there are 10 “AA” in a sequence? Look at the Nucleotides at the rows as “from” and on the column as “to” so at the

first cell you have the number we have seen in the DNA fragment going from A to A, ie AA. It is just a count, if you want to go from counts to probabilities, divide each cell number by the total sum of the row. Effectively this is a bigram model (dinucleotide model).
Do you know what is the probability for this model?

I will do an example for the first row.

from A to A -> 10/(10 + 80 + 5 + 5) = 10/100 = 0.1 A way to think about that is that the probability that we see an A base after we have seen A is 0.1.

from A to C -> 80/(10 + 80 + 5 + 5) = 0.8
Thank y I understand. So we should write all conditional probabilities?

→ Advantages and disadvantages of the models (b) and c) …
The multinomial model is simple and can be trained with less data, but it cannot capture many interesting statistics because of the strong independence assumption, i.e. the symbol of a sequence does not depend at all on the context (i.e. the symbol before it). The di-nucleotide probabilities, which is actually a 1st order Markov Chain, looks in the past to get better statistics (in our case only in the previous symbol). This, gives it more power and it can model problems as CpG island finding, but is a little more complicate than the multinomial model, since it has more parameters that need to be trained on, that is it needs more training data, so as to have good estimates.

Yes. It is the conditional probability.
Do you know how we should use this model to compute P(GACGA/M3) for the next question. Is it P(G|START) * P(A|G) * P(C|A) * P(G|C) * P(A|G) * P(END|A). P(G|START) is taken from model 2 as noted in the next question.

Shouldn't it be like that: P(G|START) * P(A|G) P(G)* P(C|A) P(A)* P(G|C) P(C) *P(A|G) * P(G)* P(END|A)?
→ No I think the right thing is the previous, but the last probability I believe should not be there, since the markov chain does not end and we do not have this probability anywhere, so the derivation would be

P(GACGA|M3) = P(G|START) * (A|G) * (C|A) * (G|C) * (A|G) = 0.1 * 0.3 * 0.8 * 0.1 * 0.3-

1.2 Which of these three models is the most appropriate for the two sequences (1) GACGA and (2) CATAT? For model 3, assume the probabilities from model 2 as initial distribution. Note: Approximate fractions where necessary so you can rank the models.

answer

You are computing the sequence likelihood under each model. We should calculate P(GACGA/M1) and same for model 2 and compare which one is bigger.

1.3 A first-order Markov model for DNA generates a nucleotide sequence: $...x_1, x_2, \ldots, x_t, x_{t+1}, \ldots$. The model is defined by states $s \in \{A, C, T, G\}$ and probabilities $p_{s_1 s_2} = P(x_t = s_2 | x_{t-1} = s_1)$ and $q_s = P(x_t = s)$. Write the following in terms of $p_{s_1 s_2}$ and q_s:

(a) $P(x_{t+1} = A, x_t = T)$

(b) $P(x_{t+1} = A, x_t = T, x_{t-1} = T)$

(c) $P(x_{t+1} = A, x_t = T | x_{t-1} = T)$

(d) $P(x_{t+1} = A, x_{t-1} = T)$

shoudn't a. be p_{TA} q_T ?

From the rules of probability we have that P(A,T) = P(A|T) P(T) thus the correct answer is

pTA qT

a. p_{TA} q_T
b. p_{TA} p_{TT} q_T
c. p_{TA} p_{TT}
d. q_A q_T

CAN SOMEONE PLEASE EXPLAIN part c??

--Do you mean part d?? About part d, since this is a Markov Chain (1st order Markov Model), P(X_t+1) is independent of P(X_t-1) thus the joint likelihood can be written as P(A,T) = P(A)P(T)

-- About question c: From the product rule of probability we have that:

P(X+1, X | X-1) = P(X+1 | X-1, X) P(X | X-1) = P (X+1 | X) P(X| X-1) since X+1 is conditionally independent of X-1 given X (Again since it is a Markov Chain).

Can we also represent c) as P(X+1, X | X-1) = P(X+1, X , X-1) / P(X-1) = Pta Ptt Qt / Qt = Pta Ptt

-- > Yes this can be done, but we need to show it by computing the expression in question (b).

So P(T|T) P(A|T) = pTT pTA

b. shouldn't this be: qT*pTT*qT*pTA*qT?

-- About question b, we have again from the product rule of probability that
P(X+1, X, X-1) = P(X+1 | X, X-1) P(X | X-1) P(X-1) = P(X+1 | X) P(X | X-1) P(X-1) from Markov property again… So:
P(A|T)P(T|T)P(T) = pTA pTT qT
I used the rule in p.9 of lecture 3 and i wrote: P(X-1)*P(X|X-1)*P(X-1) * P(X+1|X)*P(X)
-- Hey, sorry but I cannot find the rule you are mentioning in Lecture 3, but for me this does not give us the final result at least as I see it..
And from what you have written we get back P(X-1) * P(X|X-1) = P(X, X-1) the same for the other, i.e. P(X+1, X), so the final is P(X, X-1) * P(X-1) * P(X+1, X)... but I cannot easily find how this would be the same with the beginning, that is P(X+1, X | X-1)

Did you saw the product in this slide? i just replaced the joint probabilities, for example P(X,X-1)=P(X|X-1)*P(X-1)
→ Well in general the rule is the following:
P(X, Y, Z) = P(X|Y, Z) P(Y|Z) P(Z), and we can go back to see that this is the case by applying the probability rules:
P(X|Y, Z) P(Y|Z) P(Z) = P(X|Y, Z) P(Y,Z) = P(X, Y, Z)

the one you gave is right and is for two variables, and for three is the one I gave and the one that is shown in the slides in Lecture 3, the slide 'Markov Chains Sequence Probabilities', look at the second line of the equations :)

1.4 What are the limitations of a first-order Markov model when used for

(a) finding a CpG island?

(b) gene finding?

What are the limitations here? Should we mention about noise in data for example?

a. For finding a CpG island using a Markov Chain we have the issue that the CpG islands may have different lengths (which actually is the case). So your goal is to find from a big sequence n CpG islands. Using a Markov Chain, one way to do it is to use a sliding window say of 100 bp, check the statistics as it was shown in the example in the lecture, and do this for all possible sites. Then in order to see where you might have a CpG island, you have to take the log likelihood ratios for each of these windows (the beta_{xi-1, i} in the slides). This as it can be easily seen, cannot work, since a the choice of 100 bp may be good for some islands but not good modelling for others, also by performing all these comparisons it is not efficient and in many it does not work. Thus, a solution is to use the HMM as it is shown in the slides…

b. For gene finding, at least something that could be said is that the 1st order Markov Model cannot capture the statistical properties of the gene finding problem, since we need to work with codons

1.5 Describe a hidden Markov model to model a coding DNA sequence. How many parameters are required?

answer
Is this the same as asking to model a gene?

Yes, I think so. The exon model could also be the answer
Did we see a model for this on the slides?
Yes, VEIL(the exon model I was talking about earlier)

I saw in previous exams that they asked about approaches of identifying genes/proteins. What do you think we should answer to that?

I guess you could make an HMM for pretty much anything. For example for a gene you would make a state for the transcription start site, promoter, exon etc…
I don't think there is a single correct answer for that. Anyway the actual tools use higher order models that look at multiple base pairs at a time.

Thanks! I was also thinking about HMMs but what would be another approach?
Another approach would be to use other machine learning tools, SVMs is pretty popular. It works by taking kmers (little sequence fragments of length k → e.g. 4-mer = AACT or TTTG ...) where k is usually fixed.

Then it counts the number of k-mers that appear in the sequence for example

sequence = AACTGAA
let's say K=2

possible kmers : AA AC AG AT CA CC CG CT etc…
in this case the kmer counts would be:

AA = 2
AC = 1
CT = 1
TG = 1
GA = 1
all others = 0

you do this for a bunch of positive sequences and a bunch of negative sequences (label them + or -), shove it into an svm where the input is the kmer counts, the output would be positive if there's a gene or negative otherwise.

I don't remember having seen this for gene prediction but I think it could work to some extent. It's definitely been used for cancer tissue classification,protein domain classification; splice site prediction and nucleosome positioning, finding CpG islands and other things.