

# MS2a, Exercises Week 8

Rune Lyngsø

May 18, 2010

## A Hidden Markov Model Use

- a. Consider a hidden Markov model emitting sequences over the alphabet  $\{A, C, G, T\}$ . The model has two states, 0 and 1, that are equiprobable start states. Transition probabilities are 0.75 for remaining in a state and 0.25 for switching to the other state. In state 0 A and G are emitted with probability 0.45 while C and T are emitted with probability 0.05. In state 1 A and G are emitted with probability 0.05 while C and T are emitted with probability 0.45. What is the probability of observing the sequence ACTG? Observe that we do not have an end state providing explicit termination, so the model will not model a sequence length distribution. Rather, for every sequence length it models a distribution over sequence content.
- b. What is the most likely sequence of hidden states, and how probable is it?
- c. What is the most likely hidden state at position 2, summing over all possible paths, and how probable is it?

## B Hidden Markov Model Design

- a. The occasionally dishonest casino is a standard HMM example. Given the season and the problems ludomania causes in modern society, we will consider a rephrased version of this example. **Infinity Road** is an endless row of houses with a child living in most. Given the length of the street, Santa and his elves cannot constantly check the behaviour of all the children in the road, but checks just one deed per child each year to see whether it is Naughtly or Nice. However, even a Nice child may transgress and perform a Naughtly act (with probability 10% i.i.d. for all Nice children), and a Naughtly child may inadvertently find himself doing something that can be classified as Nice (with probability 5%, again i.i.d. for all Naughtly children). Santa would like to do better than basing his judgement on just

a single observation, and it just so happens that **Infinity Road** segments into Good neighbourhoods and Bad neighbourhoods, both of lengths that are geometrically distributed and with an expected length of a Good neighbourhood of 20 houses and an expected length of a Bad neighbourhood of 40 houses. All children living in a Good neighbourhood are Nice, and all children living in a Bad neighbourhood are Naughty. It is equally likely that **Infinity Road** starts with a Good neighbourhood as with a Bad neighbourhood. Houses with no children living in them are distributed uniformly at random, with on average one out of every 500 houses not having a child living in it. Neighbourhoods either side of one or more childless houses are uncorrelated, such that the neighbourhood starting after a childless house has equal chance of being Good and Bad. Design a hidden Markov model that can help Santa use the observations for all the children to annotate each child as either Naughty or Nice – don't worry that it would normally take infinitely long time to annotate an infinitely long sequence.

- b. Construct a HMM that generates the sequence  $A^i$ , i.e. the sequence of  $i$  As, with probability  $2^{-i}$  for  $i \geq 1$ , if possible. Otherwise argue it is not possible.
- c. Construct a HMM that generates sequences over the alphabet  $\{A, G\}$  with probability  $(k-1)2^{-k}$  for generating a sequence of length  $k \geq 2$ , and for which all sequences that are generated of length  $k$  are on the form  $A^i G^{k-i}$  and equiprobable.
- d. Construct a HMM that generates the sequence  $A^i G^i$  with probability  $2^{-i}$  for  $i \geq 1$ , if possible. Otherwise argue that it is not possible.

### C Stochastic Context Free Grammars

- a. Consider the context free grammar  $G$  with variables  $\{S\}$ , alphabet  $\{(, )\}$  (i.e. left and right parentheses), start variable  $S$ , and productions

$$S \rightarrow (S) \mid SS \mid ()$$

For each of the following three strings, determine whether the string can be generated from  $G$ . If the string can be generated from  $G$ , provide a derivation generating the string.

- $()()$
- $\epsilon$  (the empty string)
- $(())$

- $()()$

b. Assume that  $Pr$  assigns a probability to each of the productions of  $G$ , with

$$Pr(S \rightarrow (S)) = 0.5 \quad Pr(S \rightarrow SS) = 0.3 \quad Pr(S \rightarrow ()) = 0.2$$

What is the probability of generating the string  $()()$ ?

What is the probability of generating the string  $()()()$ ?

c. In the grammar of question a, the string  $()()$  can be derived both as  $S \Rightarrow SS \Rightarrow ()S \Rightarrow ()()$  and as  $S \Rightarrow SS \Rightarrow S() \Rightarrow ()()$ . These two derivations are essentially the same, though, the only difference is whether we choose to first replace the first  $S$  in  $SS$ , or first replace the second  $S$ . A *leftmost derivation* is one where we always replace the leftmost variable in the current string. Only the first of the above derivations is leftmost. A grammar for which we can find a string that has at least two different leftmost derivations is called *ambiguous*. For each of the following three grammars, determine whether they are ambiguous. For each ambiguous grammar, provide a string and two different leftmost derivations of that string that proves the ambiguousness.

- $G_1$  has variables  $\{S\}$ , alphabet  $\{(),\}$ , start variable  $S$ , and productions

$$S \rightarrow (S) \mid SS \mid \epsilon.$$

(remember that  $\epsilon$  denotes the empty string).

- $G_2$  has variables  $\{S, A\}$ , alphabet  $\{(),\}$ , start variable  $S$ , and productions

$$\begin{aligned} S &\rightarrow AS \mid \epsilon \\ A &\rightarrow (S). \end{aligned}$$

- $G_3$  has variables  $\{S, A\}$ , alphabet  $\{(),\}$ , start variable  $S$ , and productions

$$\begin{aligned} S &\rightarrow AS \mid A \\ A &\rightarrow (S) \mid \epsilon. \end{aligned}$$

- $G_4$  has variables  $\{W, V, I, U\}$ , alphabet  $\{l, u, r\}$ , start variable  $W$ , and productions

$$\begin{aligned} W &\rightarrow \epsilon \mid Wu \mid WV \\ V &\rightarrow lUr \mid lUVUr \mid lIlr \\ I &\rightarrow V \mid Iu \mid uI \mid II \\ U &\rightarrow \epsilon \mid uU. \end{aligned}$$

Note the close resemblance between this grammar and the recursions in equations (6)–(8) of the RNA lecture notes. An investigation into the undesirable features of ambiguity in RNA secondary structure grammars is part of [2]. There are no algorithms that for all context-free grammars can determine whether they are ambiguous, but it can be determined for large classes of context-free grammars [1].

## References

- [1] C. Brabrand, R. Giegerich, and A. Møller. Analyzing ambiguity of context-free grammars. In *Proc. CIAA '07*, volume 4783 of *LNCS*. Springer-Verlag, 2007.
- [2] R. Dowell and S. R. Eddy. Evaluation of several lightweight stochastic context-free grammars for RNA secondary structure prediction. *BMC Bioinformatics*, 5:71, 2004.