COMP3032: Intelligent Algorithms Coursework

November 3, 2011

1 Introduction

This coursework is based around Hidden Markov models and the dishonest casino problem. You must write MATLAB code to implement the HMM algorithms covered in lectures and write a short report discussing your results.

The questions will be based around two different HMMs which are given in Figure 1. In each case there are two dice, one of which is fair, and the other is loaded; in the first model the second dice is biased towards even numbers, in the second one it favours odd numbers.

2 Tasks

2.1 Task 1: Generate sequences from each model

From each model, generate a sequence of dice rolls of length `T` (which will be specified at run-time, I suggest you use 50 when testing). Store the sequences as arrays, and call the sequence generated from the first model `my_roll_one` and the sequence generated from the second model `my_roll_two`.

2.2 Task 2: Calculating the probability of a sequence

Load the file `rolls.mat`. It contains three sequences of dice throws, `roll_one`, `roll_two` and `roll_three`. Using these sequences:

- Use the forward algorithm to calculate the probability that each sequence was generated by each of the models.
- State which sequence you think was generated by which model and comment on the values you get, are any surprising?
- Do the same for `my_roll_one` and `my_roll_two`. Again, give a short comment on the probabilities you obtain.

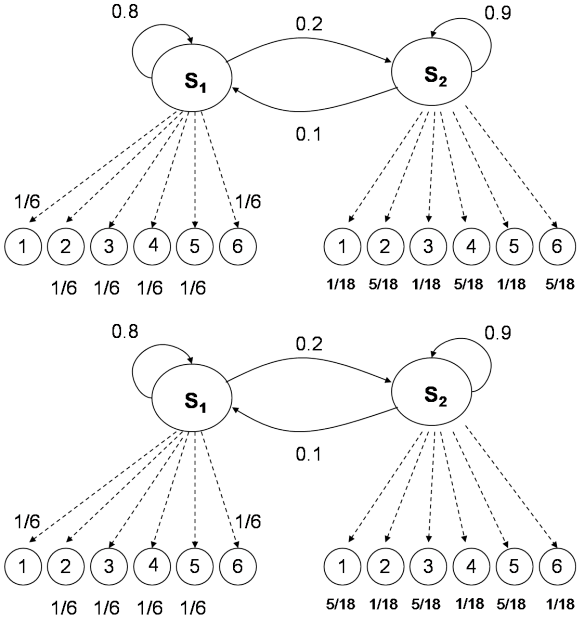

Figure 1: Two HMMs for a dishoest casino. In the top model the loaded die favours even numbers, in the bottom model the die favours odd numbers.

2.3 Task 3: Posterior Probabilities

For each of the dice rolls you must work out what the most likely hidden state is – in other words, are you more likely to be using the fair die or the loaded die. For each timestep work out whether your are most likely to be using the fair die (denote this by a 1) or the loaded die (denoted by a 2). Therefore for each roll you will store a new sequence indicating the die which was most likely being used at that time. Store the sequences in a matrix called `posterior` of size $(10, 100)$ where the 10 rows represent the sequences of dice rolls. The rows should be ordered as follows.

- Rows 1 to 5 are for model 1, rows 6 to 10 are for model 2
- Rows 1,2 (model 1) and 6,7 (model 2) are for your rolls of the dice; row 3,4,5 (model 1) and 8,9,10 (model 2) are for `roll_one`, `roll_two` and `roll_three` in that order.

Note that although T is specified at run time, you can assume that T will never be bigger than 100. If $T < 100$ then `my_roll_one` and `my_roll_two` will be shorter than the other three die rolls, in which case just pad the end of the sequence with 0's.

3 What to hand in

You need to hand in two files, a matlab script called `coursework.m` and a `comp3032.pdf` file with your findings and comments.

The matlab script must be a function:
`function [my_roll_one,my_roll_two,task_two,posterior] = coursework(T)`
which takes in the number of timesteps `T` and computes the quantities outlined above (I've provided a skeleton `.m` file for you). It MUST be named correctly (I will call it from my own script). Your function must load in the `rolls.mat` file and store results you get in the following variables:

- For task 1, it must store the dice rolls in two arrays `my_roll_one` and `my_roll_two` which are of size `(1,T)`.
- For task 2, it must store the probabilities of the sequence in on array `task_two` of size `(1,10)` with the elements representing the probabilities of generating each sequence. The probabilities must be stored in the same order as the row order specified for the state sequences above.
- For task 3, you must store your state sequences in a matrix called `posterior`.

The `.pdf` file should include the results you obtained from all of the tasks (just the probabilities I ask for, do not print out long sequences of die rolls), and answers to the questions posed in the tasks along with any other comments on the results that you obtained. You should also comment on any additional assumptions that you made, or any specific details of the coding itself that you feel are relevant.

4 Deadline

The deadline for this work is **16:00, Thursday 8th December 2011**. You must hand in the two files electronically via the C-BASS hand-in system.