BME/4800, CSE 3800/5800 — Bioinformatics — Yufeng Wu — Fall 2017
Programming Project ——- Due: 10/31/2017.

1 Overview

The goal of this project is gaining experience with the hidden Markov models (HMM). You will implement a simple HMM-based probabilistic inference program. By doing this, you get first-hand experience on how to use HMM. The particular problem you will work on is rather simple. But hopefully by trying out yourself, you get a sense of how HMM and the related algorithms work.

The problem is the example given in the textbook: the occasionally dishonest casino problem. Recall that in this problem, the casino sometimes uses a loaded dice. The problem is to study how the casino is using the two types of dices: fair and loaded. The given input is the sequences of observed results of dice rolling.

2 Tasks

This part is about the basics of HMM. You need to perform the following.

1. Model building and simulation. You need to build the HMM for modeling the casino's usage of dices. Choose some settings for the transition and emission probabilities: you can experiment your own settings. But you need to report results for the particular settings on page 55 of your textbook. Then you should support simulation of the HMM in order to generate the dice rolling output of various length (and also keep track of the sate path for later comparison). The simulation will serve as the test data generator.

2. Implement the standard algorithms for HMM: the forward, backward, Viterbi decoding and posterior decoding. I know there may exist some existing packages that already implement these algorithms. For the purpose of learning, you are required to implement these algorithms yourself.

3. Parameter estimation. Students in CSE 5800 *must* implement the Baum-Welch algorithm for parameter estimation. For students in CSE 3800 and BME 4800: you don't need to implement the Baum-Welch algorithm. Instead, you can use the simple maximum likelihood estimates as explained in the textbook. Note that you need to output the underlying states when generating simulation data.

4. Performance evaluation. You need to run your algorithms on various simulation data (of different length: 300, 3,000, 30,000 and 300,000) to answer the following questions: (a) How accurate is the parameter estimation using Baum-Welch training or the MLE approach? For this, you need to compare the trained model with the true model: what are the average errors? (b) What is the accuracy of the Viterbi decoding and posterior decoding on these simulation data? (c) What is the running time of your algorithms?

3 Hand ins

A written report including the following: (a) Problem description and overview, (b) brief description of your implementation (state any main issues you encountered during development), (c) results on accuracy of training, decoding and running time on different simulation input. Please do not repeat algorithmic details stated in the textbook.

Submit both the project report, your source code and simulation results (as a text-based log files). The submission should be performed at the HuskyCT website of this class. Do not hand in hard-copies.