732A96/TDDE15 ADVANCED MACHINE LEARNING

LAB 2: HIDDEN MARKOV MODELS

JOSE M. PEÑA
IDA, LINKÖPING UNIVERSITY, SWEDEN

1. INSTRUCTIONS

- **Deadline for individual and group reports**
  See LISAM.
- **What and how to hand in**
  Each student must send a report to LISAM with his/her solutions to the lab. The file should be named FirstName_LastName.pdf. The report must be concise but complete. It should include (i) the code implemented or the calls made to existing functions, (ii) the results of such code or calls, and (iii) explanations for (i) and (ii).

  In addition, students must discuss their lab solutions in a group. Each group must compile a collaborative report that will be used for presentation at the seminar. The report should clearly state the names of the students that participated in its compilation and a short description of how each student contributed to the report. This report should be submitted via LISAM. The group reports are corrected and graded. The individual reports are also checked, but feedback on them will not be given. A student passes the lab if the group report passes the lab and the individual report has reasonable quality, otherwise the student must complete his/her individual report by correcting the mistakes in it.

  Attendance to the seminar is obligatory. In the seminar, some groups will be responsible for presenting their group reports. Each student in these groups must be prepared to individually present an arbitrary part of the report. The selection of the presenters is done randomly during the seminar. In the seminar, some groups will act as opponents to the reports provided by the presenters. The opponent group should examine the group report of the presenter group before the seminar, and prepare a minimum of three questions, comments and/or improvements. The opponent group will ask these questions during the seminar. Check LISAM for the list of presenter and opponent groups.
- **Resources**
  The lab is designed to be solved with the R package `HMM`. Students may also want to use the RStudio development environment. To install the package in the university's lab environment, do the following:
  (1) Open a terminal window.
  (2) Write ”module add courses/732A96” or ”module add courses/TDDE15” in the terminal window.
  (3) Write ”rstudio” in the terminal window.
  - Literature:
    * Package documentation.
  - Hint: Spend the first hour reading the package documentation at `https://cran.r-project.org/web/packages/HMM/HMM.pdf`. You may even want to try the dishonest casino example included in the package.

2. Questions

The purpose of the lab is to put in practice some of the concepts covered in the lectures. To do so, you are asked to model the behavior of a robot that walks around a ring. The ring is divided into 10 sectors. At any given time point, the robot is in one of the sectors and decides with equal probability to stay in that sector or move to the next sector. You do not have direct observation of the robot. However, the robot is equipped with a tracking device that you can access. The device is not very accurate though: If the robot is in the sector i, then the device will report that the robot is in the sectors $[i-2, i+2]$ with equal probability.

(1) Build a hidden Markov model (HMM) for the scenario described above. Note that the documentation for the `initHMM` function says that the `emissionProbs` matrix should be of size [number of states]x[number of states]. This is wrong. It should be of size [number of states]x[number of symbols]. The package works correctly, though. It is just the documentation that is wrong.

(2) Simulate the HMM for 100 time steps.

(3) Discard the hidden states from the sample obtained above. Use the remaining observations to compute the filtered and smoothed probability distributions for each of the 100 time points. Compute also the most probable path.

(4) Compute the accuracy of the filtered and smoothed probability distributions, and of the most probable path. That is, compute the percentage of the true hidden states that are guessed by each method.

Hint: Note that the function `forward` in the `HMM` package returns probabilities in log scale. You may need to use the functions `exp` and `prop.table` in order to obtain a normalized probability distribution. You may also want to use the functions `apply` and `which.max` to find out the most probable states. Finally, recall that you can compare two vectors A and B elementwise as A==B, and that the function `table` will count the number of times that the different elements in a vector occur in the vector.

(5) Repeat the previous exercise with different simulated samples. In general, the smoothed distributions should be more accurate than the filtered distributions. Why ? In general, the smoothed distributions should be more accurate than the most probable paths, too. Why ?

(6) Is it always true that the later in time (i.e., the more observations you have received) the better you know where the robot is ?

Hint: You may want to compute the entropy of the filtered distributions with the function `entropy.empirical` of the package `entropy`.

(7) Consider any of the samples above of length 100. Compute the probabilities of the hidden states for the time step 101.