Manuel Haußmann, Ullrich Köthe
ullrich.koethe@iwr.uni-heidelberg.de

Exercise 7

Deadline: 11.07.2018

This exercise deals with Markov Chains and Hidden Markov Models.

Regulations

Please create a PDF file `markov.pdf` for your written answers and a Jupyter notebook `markov.ipynb` for your code (export the notebook into `markov.html` as well). Zip all files into a single archive with naming convention (sorted alphabetically by last names)

`lastname1-firstname1_lastname2-firstname2_exercise07.zip`

or (if you work in a team of three)

`lastname1-firstname1_lastname2-firstname2_lastname3-firstname3_exercise07.zip`

and upload it to Moodle before the given deadline. We will give zero points if your zip-file does not conform to the naming convention.

1 Maze (18 Points)

1.1 The way out

For this exercise we will look at a random walker inside a maze. The topology of the maze is described by the adjacency matrix M (provided in file `maze.npy` on Moodle), where $M_{ij} = 1$ means that there is a door leading from room number i to room number j (and no door if $M_{ij} = 0$). In our example M is symmetric.

Inside the maze, there is a random walker who starts in room 0 and is supposed to reach room 99. In each timestep, the walker chooses to go through one of the doors in the current room with equal probability. We can describe this process as a Markov chain.

Tasks:

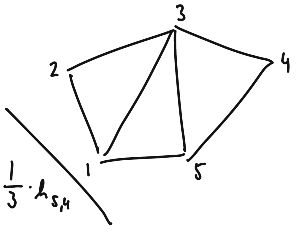
- Find the deterministic shortest path between room 0 and room 99 according to the adjacency matrix M.
- Explain the general method to derive the corresponding transition probability matrix of an arbitrary adjacency matrix M.
- Compute the transition probability matrix from the given M.
- Derive a general method to find the most likely random-walk path to get from room i to room j.
- Find the most likely path between room 0 and room 99. How much longer than the shortest path is it?

1.2 Expected traversal time

We are also interested in the expected time it takes the random walker to traverse the maze. We define the traverse time h_{ij} that describes the time to reach room j for the first time, starting at i. In general you will find $h_{ij} \neq h_{ji}$. The easiest way to calculate $h_{0,99}$ is to compute all traversing times simultaneously. You will have to find the equations yourself, but consider the following toy example:

$$h_{1,1} = 0$$

$$h_{1,4} = \underset{\substack{\uparrow \\ \text{one step to go to a neighbor of 1}}}{1} + \frac{1}{3} \cdot h_{2,4} + \frac{1}{3} \cdot h_{3,4} + \frac{1}{3} \cdot h_{5,4}$$



Tasks:

- Derive a general formula for all h_{ij}.
- Calculate the expected time it takes the random walker to go from room 0 to room 99 using the derived equation and the values from M.
- Implement the random walker and measure the average of $h_{0,99}$ over 100000 walks.

2 Chimpanzee (13 Points)

The lecture introduced a movement model for a chimpanzee in the jungle. We assume here that the transition and observation probabilities are already known. Learning will be addressed in the next homework. The possible hidden states of the chimpanzee are "north of the river" =1 and "south of the river" =2. We know that he stays in only one location per day, but don't know how he decides to switch. Therefore we assume a transition matrix

$$A = \begin{pmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{pmatrix}$$

Also we cannot spot him with absolute certainty. The following 3 spotting results are possible for a given day:

0. chimp was not spotted, or observations contradict
1. chimp was believed to be seen in the north
2. chimp was believed to be seen in the south

The transition matrix from the hidden states to the measurements is

$$B = \begin{pmatrix} 0.5 & 0.4 & 0.1 \\ 0.4 & 0.1 & 0.5 \end{pmatrix}.$$

We can interpret the entries in B as follows: B_{ij} is the probability that the chimp is seen in location j when he is actually in i.

Tasks:

- Create your own Python implementation of the forward-backward and Viterbi algorithms.
- Construct an HMM matching the probabilities above and find the globally most probable sequence of hidden states
$$\widehat{\mathbf{x}} = \text{argmax}_{\mathbf{x}} \log(p(\mathbf{x}, \mathbf{y} = \mathbf{o}))$$
for the observation sequence
$$\mathbf{o} = [0, 1, 1, 0, 2, 0, 2, 2, 2, 0, 2, 2, 2, 2, 2, 0, 0, 1, 1, 2]$$
- Compute the individual marginal probabilities $p(x_j|\mathbf{y} = \mathbf{o})$ and check if the sequence of their maxima coincides with the global MAP solution from the previous task.

3 Pairwise Marginals (9 Points)

The lectureexplained the Baum-Welch algorithm for learning the transition probabilities from a HMM for a given training sequence $\mathbf{y} = \mathbf{o}$ with unknown true hidden states $\mathbf{x}$. The algorithm relies on various expressions for marginal probabilities. The pairwise marginal was used but not derived in the lecture.

Task:

- Derive a recursive rule to compute $q(x_{j-1}, x_j, \mathbf{y} = \mathbf{o})$, the unnormalized version of the pairwise marginal probability $p(x_{j-1}, x_j|\mathbf{y} = \mathbf{o})$, by means of a modified forward-backward algorithm.