Introduction to Probabilistic Graphical Models
Practical Session 1

Umut Şimşekli
Télécom ParisTech, Université Paris-Saclay, Paris, France
`umut.simsekli@telecom-paristech.fr`

Instructions: The codes should be written in Python. You can work in groups of 2. Put all your files (code and report – could be a single notebook file) in a zip file and submit it through moodle before October 15 2019, 23:50. Late submissions will not be accepted.

Question 0

In many practical applications, we often need to compute $s = \log \sum_{i=1}^{I} \exp(v_i)$, where each $v_i < 0$ and $|v_i|$ is very large. Derive (mathematically) and implement a *numerically stable* algorithm for computing 'log(sum(exp(v)))', where $v = \{v_i\}_{i=1}^{I}$ is a vector of numbers. Explain why it should work. Test your algorithm on log(sum(exp{-1234, -1235})).

Question 1

A robot is moving across a circular corridor. We assume that the possible positions of the robot is a discrete set with N locations. The initial position of the robot is unknown and assumed to be uniformly distributed. At each step k, the robot stays where it is with probability ϵ, or moves to the next point in counter-clock direction with probability $1-\epsilon$. At each step k, the robot can observe its true position with probability w. With probability $1-w$, the position sensor fails and gives a measurement that is independent from the true position (uniformly distributed).

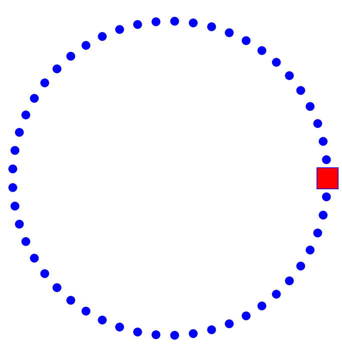

Figure 1: Robot (Square) moving in a circular corridor. Small circles denote the possible N locations.

1. Choose the appropriate random variables, define their domains, write down the generative model and draw the associated directed graphical model.

2. Define the conditional probability tables (i.e., the transition model and the observation model) given the verbal description above.

3. Specify the following verbal statements in terms of posterior quantities using mathematical notation. For example "the distribution of the robot's location two time steps later given its current position at time k" should be answered as $p(x_{k+2}|x_k)$.

   (a) Distribution of the robot's current position given the observations so far,

   (b) Distribution of the robot's position at time step k given all the observations,

   (c) Distribution of the robot's next position given the observations so far,

   (d) Distribution of the robot's next sensor reading given the observations so far,

   (e) Distribution of the robot's initial position given observations so far,

   (f) Most likely current position of the robot given the observations so far,

   (g) Most likely trajectory taken by the robot from the start until now given the observations so far.

4. Implement a program that simulates this scenario; i.e., generates realizations from the movements of the robot and the associated sensor readings. You can use the randgen function you wrote earlier. Simulate a scenario for $k = 1, \ldots, 100$ with $N = 50$, $\epsilon = 0.3$, $w = 0.8$.

5. Implement the Forward-Backward algorithm for computing the quantities defined in 3-a,b,f. (Attention: be careful with numerical stability! Modify the log-sum-exp trick for this algorithm)

6. Assume now that at each step the robot can be kidnapped with probability κ. If the robot is kidnapped its new position is independent from its previous position and is uniformly distributed. Repeat 4 and 5 for this new model with $\kappa = 0.1$. Can you reuse your code?