MACHINE LEARNING.
Assignment 5.

Due: December 12, 2017.

1. [35 points] **Playing with PCA**

   (a) [5 points] Generate 200 examples from a Gaussian with mean (5, 20) and covariance matrix:

   $$\begin{bmatrix} 10 & 2 \\ 2 & 5 \end{bmatrix}$$

   Plot the data you generated. Make a prediction about what directions the principal components should have, based on the class notes.

   (b) [10 points] Subtract the mean from all the data points and run PCA on this data. Plot the data and the principal components and comment on your result.

   (c) [10 points] Now subtract the mean from all the data points and divide the result by the standard deviation (so as to normalize the data). Run PCA again and explain what happens to the result.

   (d) [10 points] Multiply the second coordinate of every point in the data set by 1000 and run PCA again, under the two scenarios above What happens to the principle components, and why? Comment on the robustness of PCA wrt the scaling of the data.

2. [20 points] **Bayes nets**

   The starship Enterprise is under fire from the Borg and Data the android is trying to save it. He knows that if the ship is close to the Borg vessel and the power is high, he will hit the Borg with probability 90%. If the Enterprise is close but the power is low, the probability drops to 70%. If the Enterprise is far and the battery is high, the chances of a successful hit are 50%. If the Enterprise is far and the battery is low, the chance of success is only 10%. Data has a sensor which tells him whether the Enterprise is far or close, but this sensor sometimes malfunctions: it indicates the correct distance only with probability 80%.

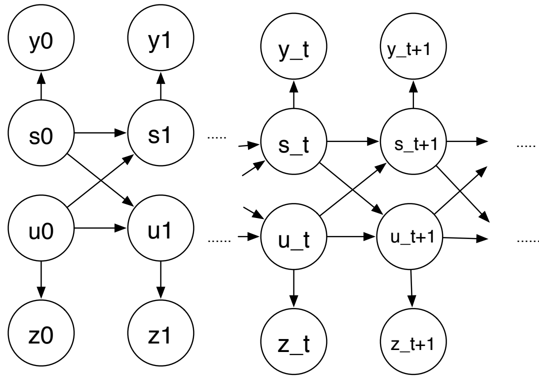
   (a) [10 points] Draw the Bayesian network corresponding to this problem (including the probability tables).

   (b) [10 points] Data observes that the power is low and that the sensor tells him the Enterprise is close. Compute the probability that he will be able to hit the Borg.

3. [45 points] **Coupled Hidden Markov Models**

   In the class notes there are several models for reasoning with sequences of data (trajectories). The HMM is the simplest such example, in which states are hidden, and we see observations that depend on the state. The Coupled Hidden Markov model (CHMM) is a similar kind of graphical model: we have several hidden Markov models running in parallel, and their states interact. This model is quite useful, for example, when you try to parse video, and you consider the observations as being sound and visual data, respectively.

   Consider a system with two HMMs, depicted in Figure 1:



Here, s_i and u_i are the states of the two coupled HMMs, y_i and z_i are the observations coming from the two chains, and the two chains interact in the way depicted in the picture.

(a) [5 points] Specify what are the parameters of this model.

(b) [10 points] Derive an algorithm for computing the joint probability of a sequence of observations $(y_0, z_0), (y_1, z_1) \dots (y_T, z_T)$.

(c) [10 points] Derive a forward algorithm that computes the most likely sequence of hidden states given a sequence of observations. You recall that in order to do this, in the case of a simple HMM, you maintain a "belief state", which gives the probability of each hidden state based on the observations seen so far. You can use a similar idea here. Alternatively, you may consider how you can apply the junction tree algorithm to this situation.

(d) [10 points] Suppose that instead of the chains being coupled at every time step, the coupling only happens every k time steps (on time step 0, k $2k$ etc). For $k = 1$, you get the same model as above. If k is fairly large compared to the length of sequences, the chains are called **loosely coupled**. Describe how your model and the inference algorithms change in this case.

(e) [10 points] Suppose that you observe several sequences of two time series and you know that they come from a loosely coupled HMM; you know the number of possible states for each individual chain, but you do not know k. Describe a learning algorithm for this problem.